1977-06

Evaluation of and operational procedures for a helicopter simulation system utilizing an integrated electronic instrument display.

Fetzer, William Woodrow, Jr.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/18115

EVALUATION OF AND OPERATIONAL PROCEDURES FOR A HELICOPTER SIMULATION SYSTEM UTILIZING AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY

William Woodrow Fetzer

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Evaluation of and Operational Procedures
For a Helicopter Simulation System
Utilizing an Integrated Electronic
Instrument Display

by

William Woodrow Fetzer, Jr.

June 1977

Thesis Advisor: D. M. Layton

Approved for public release; distribution unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Evaluation of and Operational Procedures for a Helicopter Simulation System Utilizing an Integrated Electronic Instrument Display		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1977 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) William Woodrow Fetzer, Jr.		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1977 13. NUMBER OF PAGES 88
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

instrument flight
helicopter
integrated display

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report discusses the evaluation and documentation of an integrated electronic instrument display designed to investigate stability and control of a helicopter during precision hover operations. The equations of motion, developed from the Kaman SH-2F Seasprite helicopter, were implemented by a hybrid computer system and displayed by a graphics processor. A complete procedural checklist, including troubleshooting methods,

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601 |

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

is included in this report. This helicopter simulation system can be used for further research in the development of optimal heads-up display configurations as well as analyses of instability caused by pilot induced oscillations in the hover flight regime.

Approved for public release; distribution unlimited

EVALUATION OF AND OPERATIONAL PROCEDURES FOR A HELICOPTER SIMULATION SYSTEM UTILIZING AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY

by

William Woodrow Fetzer, Jr.
Lieutenant, United States Navy
B.S., United States Naval Academy, 1970

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN AERONAUTICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1977

ABSTRACT

This report discusses the evaluation and documentation of an integrated electronic instrument display designed to investigate stability and control of a helicopter during precision hover operations. The equations of motion, developed from the Kaman SH-2F Seasprite helicopter, were implemented by a hybrid computer system and displayed by a graphics processor. A complete procedural checklist, including troubleshooting methods, is included in this report. This helicopter simulation system can be used for further research in the development of optimal heads-up display configurations as well as analyses of instability caused by pilot induced oscillations in the hover flight regime.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The rapid advancement in computer technology has greatly improved the development of visual heads-up displays for aircraft instrumentation. Microprocessor and graphic technology coupled with radar altimeter and Doppler radar systems inputs enable designers to create various tactical and navigational displays. Perhaps the most difficult task experienced by today's helicopter pilot is hovering in instrument flight conditions at night. Precision hovering at low altitude and low airspeeds requires the utmost skill in instrument flying, and the display systems in present-day helicopters are less than adequate.

Reference 1 cited development efforts directed in two major areas relative to the precision hover task; increased stability and control systems, and navigational displays to provide hover information over a fixed position. Among the concepts to be examined were a low airspeed sensor system and integration of a forward looking infrared radar (FLIR) with a CRT-fiber optic display.

With the advent of these new systems, it is important that a display scheme be developed to implement the sensor information into a suitable arrangement for precision hovering.

The helicopter simulation system described in this report does not utilize the latest advances in computer technology.

However, the basic principles of operation and interface are the same. The helicopter pilot controls the flight of the aircraft using conventional control signals responding to artificial sensor information. This sensor information is provided by a hybrid computer system which solves the six degree-of-freedom equations of motion and applies them in response to initial conditions and pilot controlled inputs.

The sensor information is relayed to a graphics computer which is used as the integrated heads-up display in the cockpit.

This simulation model has the flexibility to allow the student to investigate various display modes for feasibility and operability. In addition, stability and control analysis of the equations of motion can be examined for optimal control in the precision hover mode.

II. OBJECTIVE

The basic purposes of this work were to place the heli-computer simulator in full working condition, provide detailed operating procedures to allow further research into the simulation and the heads-up display system, and document trouble-shooting techniques and computer operations peculiar to the helicopter simulation in order to facilitate future research using this system.

III. BACKGROUND

The present helicopter simulation system is essentially the work of many students. Conversion of the C-11B Instrument Flight Trainer into a Variable Stability Flight Simulator was accomplished by Sweeney and reported in Ref. 2. Further work on the equations of motion was conducted by Huckemeyer [Ref. 3].

The equations of motion for the helicopter were initially developed by Hoxie and are contained in Ref. 4. The analog patchboard implementing the equations of motion and computer interfacing were also done originally by Hoxie. The analog diagrams are provided in Appendix A for easy reference. The display arrangement and the digital computer program are based on the work of Ammerman [Ref. 5]. A complete listing of the digital program is provided in Appendix B, and the corresponding Fortran variables are listed in Appendix C. Small changes in the analog program and the digital program were made to correct scaling errors and typographical errors. In addition, some components in the Analog computer had to be changed because of hardware failures.

IV. THE SIMULATION

A. DESCRIPTION

The object of the simulation run is to fly to a designated hover position from a given initial position, altitude and airspeed, hover over the position for a specific time and depart in a straight-ahead climb. For each run the starting position is set by the digital computer program with the initial conditions as specified in Table I. A glossary of terms may be found in Appendix C.

As the helicopter nears the designated hover position, crew directions are provided by the Crew Direction subroutine of the digital program. The Crew Direction subroutine functions as the "eyes" of the helicopter. The displayed messages simulate actual crew commands that are used for hovering operations where the pilot is unable to see the hover target. A listing of the displayed text commands are included in Table II along with the parameters which cause the output of the associated messages. The top portion of Figure 1 shows the display with the helicopter in level flight at an altitude of 500 feet and an airspeed of 70 knots. The bottom picture of Figure 1 shows how the display appears in a 40 foot hover over the target.

TABLE I

INITIAL CONDITIONS

U	70 KTS
V	0
W	5.26 ft/sec.
θ	2.55
ϕ	0
ψ	0
X_E	-4000 yds.
Y_E	0
Z_E	500 ft.
100 B_{1c}	1.7687
200 $\Delta_{\theta_{\dot{c}}}$	-.9034

TABLE II

CREW DIRECTIONS

TARGET IN SIGHT	XE > -2000 YDS
STEADY FORWARD	XE > -2000 YDS
EASY FORWARD	XE > -30 YDS
STOP FORWARD	VX > .3 XE
STEADY HOVER	3 YDS ≤ XE ≤ 3 YDS
STOP BACK	VX < -.3 XE
EASY BACK	XE > 20 YDS
TARGET LOST	XE > 60 YDS
WAVE OFF	XE > 60 YDS
EASY LEFT	YE < 25 YDS
STOP LEFT	VY < -.3 YE
EASY RIGHT	YE < -25 YDS
STOP RIGHT	VY > .3 YE
MAN ON HOISE	HOVER TIME > 120 SEC
MAN IN AIRCRAFT	HOVER TIME > 150 SEC
PULL UP YOU ARE LOW	ZE < 15 FT

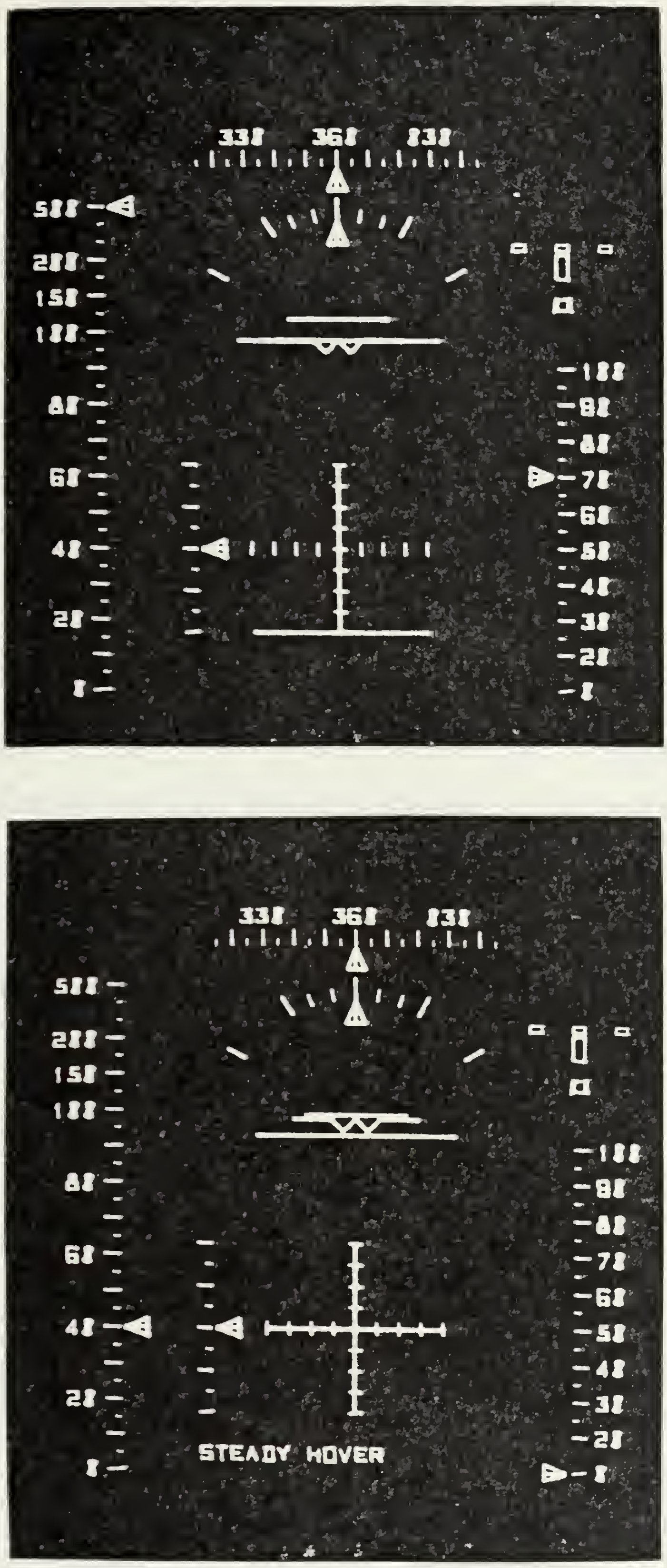


Figure 1. Integrated Electronic Instrument Display

B. COMPUTER FACILITIES LAYOUT

The helicopter simulation system is comprised of three main computing systems which are described in detail in Ref. 6. Each of these components (shown in Figure 2) is an integral part of the system, linked by a network of interfaces and trunklines. In addition, a television camera, placed in front of the appropriate graphics display console, is linked by cable to a television repeater in the cockpit.

C. AIRCRAFT DYNAMICS

The stability derivatives and equations of motion used for this simulator are taken from the SH-2F "Seasprite" helicopter produced by the Kaman Aerospace Corporation. The equations of motion were developed by Hoxie [Ref. 4] and modified slightly by Ammerman [Ref. 5]. The normalized stability derivatives were supplied by Kaman and are listed in Table III.

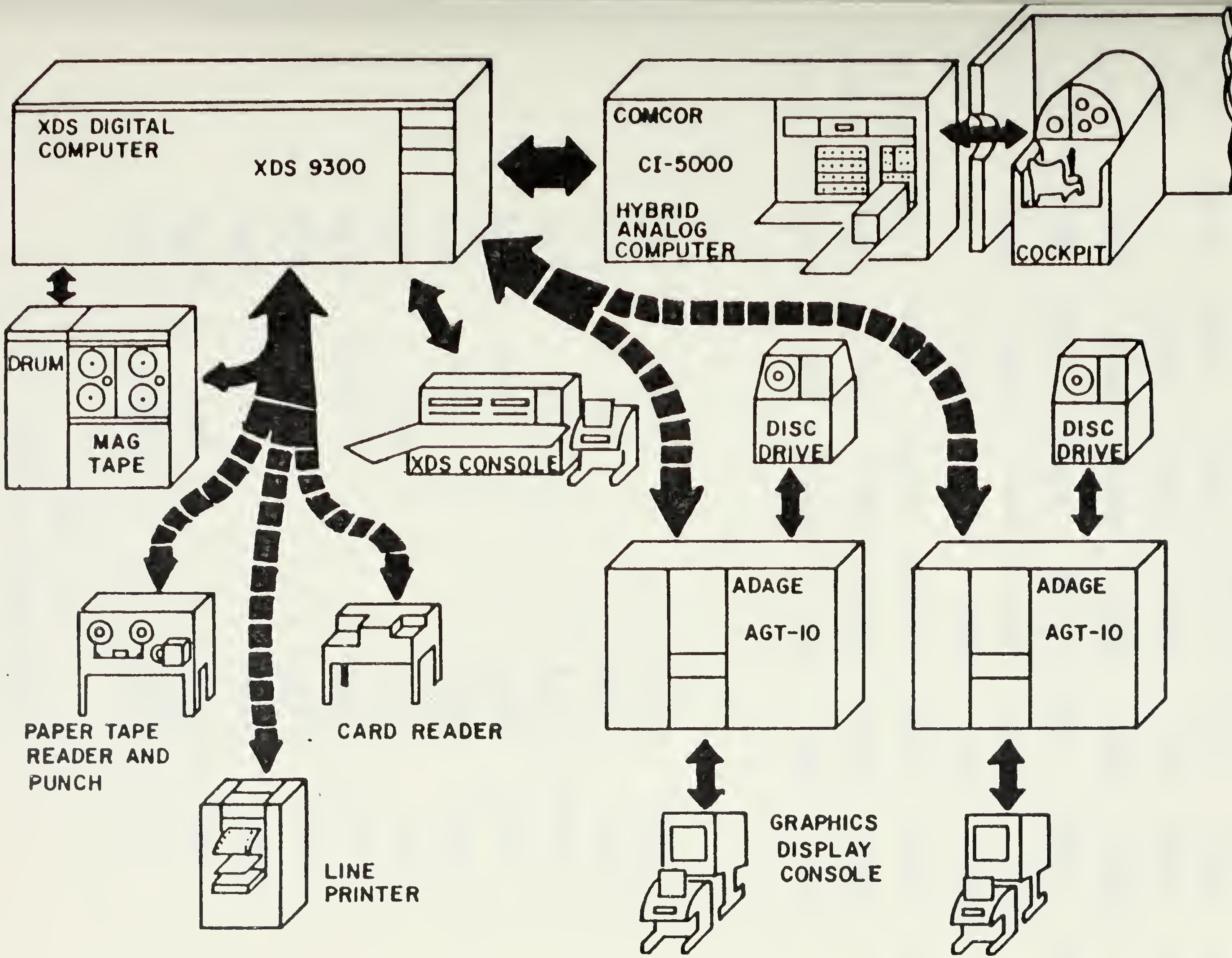


Figure 2. Naval Postgraduate School Computer Laboratory

	0 KTS.	30 KTS.	50 KTS.	70 KTS.	91 KTS.	112 KTS.	136 KTS.
$X_A(u)$	0	.2328	-1.501	-3.814	-6.847	-10.62	-15.51
$X_{\theta_c}(u)$	25.06	22.88	29.96	19.94	17.55	17.62	20.45
$Z_A(u)$	0	-9.272	-5.854	.6023	6.554	18.76	34.34
$Z_w(u)$	-.4045	-.5092	-.5843	-.6368	-.6682	-.6875	-.6928
$Z_{B_{1c}}(u)$	4.567	35.35	64.79	97.41	131.1	164.4	197.4
$L_v(u)$	-.0215	-.0261	-.0298	-.0352	-.0409	-.0464	-.0519
$M_A(u)$	0	.1417	.1940	.2244	.2610	.2456	.1835
$M_{B_{1c}}(u)$	-12.17	-12.22	-12.34	-12.54	-12.82	-13.04	-12.15
$N_r(u)$	-.5871	-.7410	-.8881	-1.080	-1.269	-1.447	-1.622
$N_v(u)$	.0172	.0202	.0227	.0272	.0312	.0352	.0399

$X_A(0)$	2.806	$Y_p(0)$	-1.139
$X_q(0)$	.8689	$Y_r(0)$	.9627
$X_w(0)$	.0491	$\bar{Y}_{A_{1c}}(0)$	42.63
$\bar{X}_{B_{1c}}(0)$	40.13	$Y_{\theta_r}(0)$	18.16
$Z_A(0)$	-32.08	$L_p(0)$	-2.425
$Z_q(0)$	.5228	$L_r(0)$	.4082
$\bar{Z}_{\theta_c}(0)$	-208.2	$\bar{L}_{A_{1c}}(0)$	36.51
$M_q(0)$	-.7853	$L_{\theta_r}(0)$	7.075
$M_w(0)$	-.0002	$N_p(0)$	-.0072
$M_{\theta_c}(0)$	.7789	$\bar{N}_{A_{1c}}(0)$	1.877
$Y_v(0)$	-.0338	$N_{\theta_r}(0)$	-11.86

TABLE III

STABILITY DERIVATIVES

V. OPERATING PROCEDURES

A. INTRODUCTION

A system operating checklist for the Fixed-Base helicopter simulator is presented in Appendix D. A copy of this checklist is also located in the simulator cockpit. The basic operating procedures for each individual computing system are presented in Ref. 6. However, there are no specific operating instructions for the helicopter simulation system which includes items peculiar to the simulator. This section includes a step by step discussion of the procedures necessary for simulator operation. The system could be run entirely from this presentation. However, the operator should carefully read the start-up and shut-down procedures [Ref. 6] for each individual system prior to attempting the first run. In addition, the assistance of a laboratory technician should be requested for a familiarization run.

B. ANALOG COMPUTER PROCEDURES

1. Install patch boards (numbered 8) on the COMCOR CI-5000 Analog Computer. The patch boards are located in cabinets behind the CI-5000 mainframe. Make sure that the guide rollers are positioned properly, and place the latching handles in the vertical position.

2. Turn the CI-5000 power switch on. The switch is located in the lower left corner of the operators display console.

3. Set potentionmeter (POT) 400 to +20.00 volts. This POT is set manually using the appropriate calibrated dial located to the left of the analog patch board. To monitor the POT setting, the following procedure must be followed using the operator's keyboard directly in front of the display console:

a. Press KEY BOARD and POTSET mode switches in sequence.

b. Press POT PS class switch.

c. Press the desired numbers in sequence for the proper address. P400 should appear in the address window in the upper left corner of the display console. The value of the potentiometer setting will appear in the RATIOMETER window of the display console. The manual dial is then rotated until +20.00 appears on the ratiometer.

4. Set POT 401 to +20.00 volts. This is accomplished in much the same manner as in the previous step. However, pressing the ADV key of the address keyboard will advance the potentionmeter address to P401. The appropriate manual dial is then rotated until +20.00 appears on the ratiometer.

5. Set POT 437 to +30.00 volts. Press 437 on the address keyboard and proceed as in the previous step.

6. Set limiters L00 and L07 to $\pm$ 1.0 volts. The limiters are set in a similar fashion to the manual potentiometers. The limiter setting dials are located above the handset potentiometers. In order to monitor the limiter settings, the following steps must be followed:

a. Press AMP on the operators keyboard.

b. Press the appropriate amplifier address (002 for L00 and 060 for L07). A002 should appear in the address window.

Adjust the limiter by pressing the center limiter switch toward the positive dial. While holding the switch in the positive direction, turn the dial until +01.00 appears in the ratiometer window of the operators display console. Then move the switch to the negative side, and obtain a value of -01.00 in the ratiometer window. Proceed in a similar manner to adjust limiter L07. Amplifier A060 is used in conjunction with L07.

7. Center all Digital Function Switches. These switches are located between the analog patchboard and the logic patchboard. The Digital Function Switches are parallel to the cockpit switches and can be used to control the simulation from the CI5000 control console. Centering these switches prevents inadvertant control signals from entering the computer.

8. The CI-5000 Analog Computer System is now properly set-up. Press the DIGITAL CMPTR mode switch which links the CI-5000 Analog Computer to the XDS-9300 Digital

computer. When the DIGITAL CMPTR mode is selected, no controls from the operators keyboard will be received. To use the operators keyboard for manual potsetting or for addressing amplifiers, pots or trunklines, the KEYBOARD key must be pressed to place the analog computer in a "stand alone mode."

C. GRAPHICS COMPUTER PROCEDURES

1. Turn on the XDS-9300 computer. Before the Adage AGT-10 graphics computer can be linked to the XDS-9300 digital computer, the XDS-9300 must be energized. Complete start-up procedures for the XDS-9300 are included in Ref. 6. However, to simply turn it on, press the RESET and POWER switches simultaneously. The XDS-9300 usually remains on during normal working hours, therefore, this step may not be necessary.

2. Place the OLD AMOS discs on the appropriate AGT-10 disc drive. The disc drives are located in the northeast corner of the computer laboratory, and the discs can be found in an adjacent cabinet. The serial numbers corresponding to the OLD AMOS discs are posted on each disc drive along with the installation instructions.

3. Turn the disc drive on by pressing the POWER ON/START switch. When the disc drive attains the proper operating RPM, the READY light should come on.

4. Turn on the circuit breaker located on the back of the appropriate AGT-10 mainframe. The circuit breaker is identified by a "This is it" label.

5. At the front of the AGT-10 mainframe is the Operator's Control Panel (OCP). Press the following control switches in sequence:

   a. HALT

   b. RESET

   c. RUN

   d. PULSE 1

6. Upon activation of the PULSE 1 switch, the teletypewriter (TTY) should type MO/DA/YR on the TTY located in front of the desired display console. If this message does not appear, follow the bootstrap loading instructions attached to the AGT-10 OCP.

7. If the white BREAK light is on at the TTY, press the red BRK RLS key prior to typing the date. Type 7/7/77 on the TTY and press the return button.

8. Type RESET ("GATED", 101)! on the TTY. The disc drive will then cycle and the TTY carriage will return. A memory map should be displayed on the CRT.

9. After the carriage returns, type GATED! This causes the program "GATED" to be executed. GATED performs three functions. It refreshes the display, communicates with the operator, and communicates with the graphics subroutines in the XDS-9300.

10. To determine if GATED has been loaded and is executing properly, press function switch "1" located in the upper left corner of the ADAGE function switch keyboard. This switch corresponds to the TEXT EDIT switch of the

GATED overlay shown in Figure 3. The message, "TEXT BLOCK SELECT MODE BLOCK 1," should be displayed on the lower ledge of the screen. If this text line does not appear, verify that the XDS-9300 is on, return to step 5 and continue from there.

D. DIGITAL COMPUTER PROCEDURES

1. Load the "HELO SIMULATION" tape on either of the two SDS tape drives. The tapes are stored in row three of the tape file cabinet located to the left of the XDS-9300 mainframe. Tape number two contains a core dump of the program contained in Appendix B. Tape number one is the same program except that PSIDTS scaling has been changed to equal 28.65 PSIDOT. Mount the tape by following the threading guide located on the tape drive doors. Refer to Ref. 6, Section I, Part II for additional instructions.

2. Run the tape forward to the load point. This is accomplished with the mode selector in the MANUAL READ position. Press and release the FORWARD DRIVE switch. The tape should move forward until the LOAD POINT light comes on.

3. Set the Mode Selector switch to AUTOMATIC. The UNIT READY light should come on.

4. Set the Tape Unit Selector switch to 1. The tape drive is now properly set up for control by the XDS-9300.

5. Place the tape rerun deck and data cards into the card reader. This deck, consisting of approximately 30

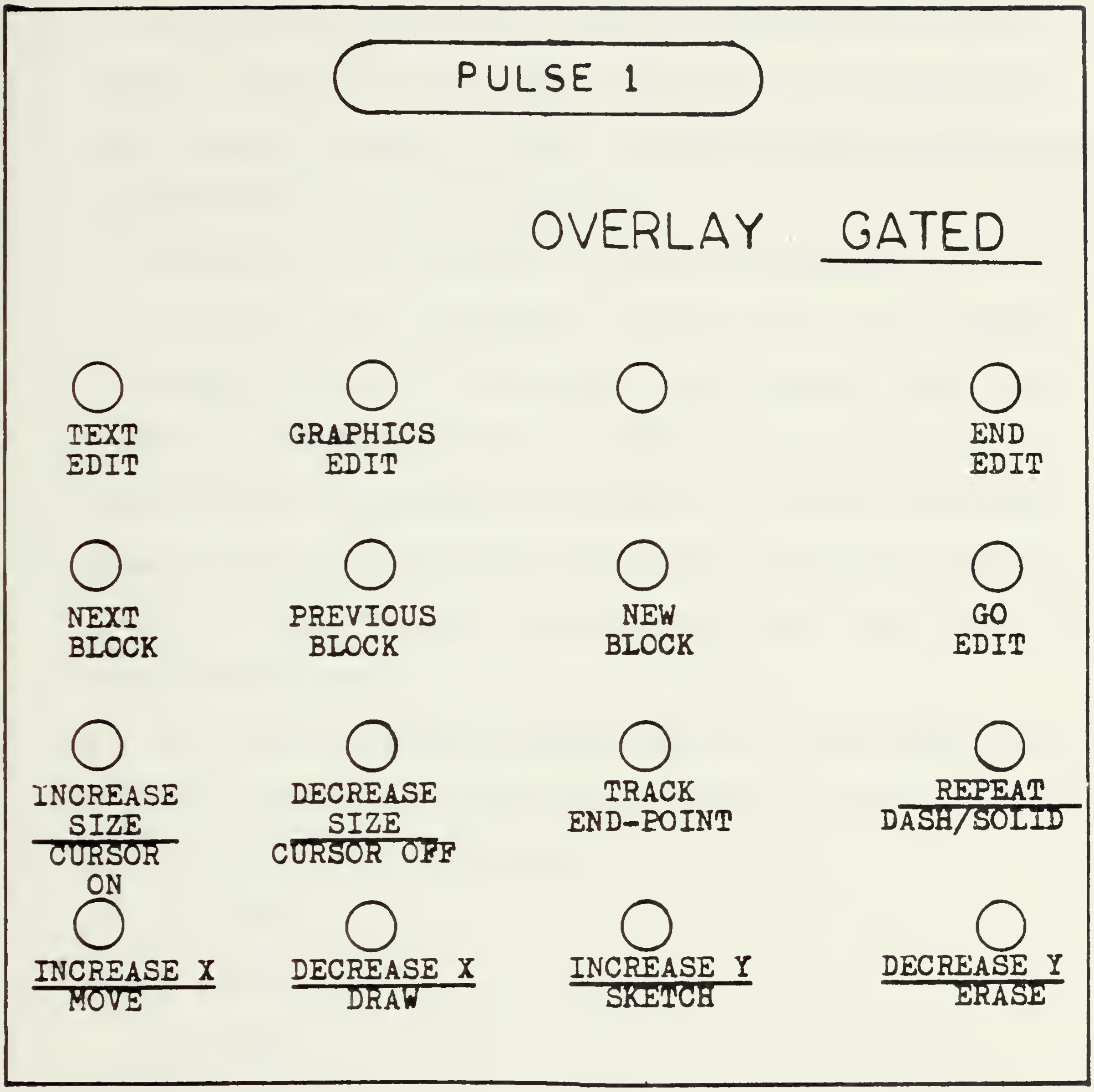


Figure 3. Function Switches (GATED Overlay Shown)

cards, is located in the card file marked HELO SIMILATION. Execution of this deck causes the core dump to be placed into the XDS-9300 along with the data.

6. Press the POWER ON and START switches on the card reader. The NOT READY light should go out, indicating that the card reader is ready to operate upon command from the XDS-9300 control console.

7. Check the line printer to see if the READY light is on. If not, press the READY switch on the line printer.

8. Select EXT on the XDS-9300 clock switch. The clock switch is located adjacent to the bottom row of circuit cards inside the number two panel of the XDS-9300 main-frame. The EXT position allows the timing frequency to conform to the frequency patched into the logic board of the CI-5000 computer.

9. To load the digital program and link the three main computer systems, the following switches on the XDS-9300 operator control console must be pressed.

   a. IDLE

   b. RESET

   c. RUN

   d. CARDS

After executing the above sequence of instructions, the following things should occur; the card reader should begin feeding cards, the tape should be read into the XDS-9300, and the potsetting routine should begin setting the CI-5000 POTS automatically. The address window of

the CI-5000 will cycle through the POTS as they are set. POTS which cannot be accurately set by the POTSET subroutine will be listed as error messages on the XDS-9300 TTY. Those POTS listed in the error messages must be set manually.

10. At the completion of the POTSET subroutine a message will appear on the TTY requesting an input for the desired AGT-10 system to be used. The appropriate AGT-10 can be selected by typing: IDEV = 1* or 2*. Press the RETURN key of the TTY. The integrated display pictured in Figure 1 should appear on the selected AGT-10 display console. A quick check, to see that the system is operating correctly, can be made by moving the Digital Function switch, DS1, to the upper position. The pointer needles on the airspeed scale and VSI should begin to move. Return the display to start by pressing the following Digital Function switches in the order listed.

 a. DS2 - down momentarily

 b. DS1 - center

 c. DS4 - down momentarily

11. Set up the TV camera. Position the camera in front of the AGT-10 screen. Attach the transmission cable to the VIDEO jack of the camera and plug the camera into the extension outlet. The transmission cable from the cockpit is located on the floor between the two AGT-10 consoles and is marked HELO SIMULATOR. Remove the lens cap and adjust the camera to the proper position.

The TV camera may be used for other projects in the computer lab. Therefore, it may be necessary to readjust the various controls on the camera for proper signal transmission. This can be accomplished by removing the TV repeater from the cockpit and bringing it into the main laboratory. A small jumper cable is located in the HELO SIMULATION card file drawer to permit direct hookup from the TV repeater to the TV camera. Adjust the camera as necessary to obtain the best picture, and mark the camera's position on the floor to preclude time consuming set-up requirements for subsequent operations.

E. COCKPIT SET-UP PROCEDURES

1. Set the FLY switch to the down position. This switch is located on the right side console of the cockpit.
2. Set the instrument display switch to INTEGRATED. The NORMAL position is for use of the regular conventional instrumentation [Ref. 5].
3. Turn on the MASTER POWER, FLIGHT SYSTEM and DC POWER SUPPLY switches located at the rear of the cockpit. The circuit breaker located on the wall to the right of the cockpit may need to be turned on.
4. Close the latching mechanism on the cockpit terminal patch board located at the rear of the cockpit. This patch board links the control potentionmeter signals to the appropriate trunklines for input to the CI-5000 computer.

5. Turn on the TV repeater in the cockpit. The simulator is now ready for operation. Each simulation run can be controlled in the cockpit or at the CI-5000 console by activating the following switches in sequence:

a. FLY (DS1) switch on

b. STOP RUN (DS2) switch on momentarily

c. FLY (DS1) switch off

d. RERUN (DS4) switch on momentarily

VI. SIMULATION SYSTEM TROUBLESHOOTING PROCEDURES

A. INTRODUCTION

Many of the problems associated with the unsuccessful operation of the helicopter simulation system can be eliminated by careful adherence to the operating instructions and procedures contained in Section V and in Ref. 6. Most of the errors relating to the specific computer system involved can be corrected by referring to Ref. 6. The problems presented in this section are those which occurred most frequently during the helicopter simulation and those not covered specifically in Ref. 6.

B. ANALOG COMPUTER PROBLEMS

1. MANUAL POTSETTING

There are several POTS which cannot be set accurately by the digital POTSET subroutine. Most frequently, those POTS are P000, P021, P035 and P042. After the POTSET subroutine finishes, press the KEYBOARD switch on the CI-5000 mode control keyboard. Press POTSET nad POT PS. Enter the desired address using the address keys. The address should appear in the ADDRESS window and the present value should appear in the RATIOMETER window. Press "+" and the desired four-digit POT value on the address keyboard. Then press the SERVO key. The POT value placed in the REFERENCE DAC should now appear in

the RATIOMETER window. If this method does not work, press the POT ADJUST key, and adjust the POT up or down by moving the SERVO SYSTEM POT switch located on the right side of the CI-5000 display console. This switch is very sensitive, therefore, careful movement is necessary to obtain the desired POT value. Press DIGITAL CMPTR to return control to XDS-9300.

2. OSCILLATING INSTRUMENT DISPLAY

If the display becomes unstable and oscillatory upon activation of the FLY (DS1) switch, the problem could be one of several hardware failures. The assistance of a laboratory technician will be needed. A problem in the addressing between the A/D or D/A converters could be present. Failure of the "conditional ground" of the CI-5000 caused unstable start-up oscillations in earlier simulation runs, requiring extensive troubleshooting by the lab technicians.

3. AMPLIFIER OVERLOADS

Overloaded amplifiers will cause incorrect signals to be present in the dynamic equations of motion. The overload condition is indicated by a light on the CI-5000 display console. The overload can be cleared by entering the keyboard mode and pressing POTSET. If the amplifier overload cannot be corrected, additional troubleshooting of the suspected amplifier inputs may be necessary. The amplifiers may fail because of hardware circuit board problems. Therefore, the overloaded amplifier should be

checked for hardware failure prior to extensive patch board trouble shooting. Other patch board problems could be caused by partial pin insertion, open wires, or bent contacts behind the patch boards.

C. GRAPHICS COMPUTER PROBLEMS

1. GATED NOT EXECUTED

Sometimes it is difficult to obtain proper execution of GATED during the graphics computer set-up sequence. If the message "FILE NOT DEFINED" appears on the graphics TTY after typing GATED!, Press PULSE 1 and retype GATED!. If GATED is not executed after several attempts, ask a laboratory technician for assistance.

2. DISPLAY FAILURE DURING SIMULATION RUN

Occasionally the CRT display will go blank during a simulation run. Retype GATED!, and continue the start-up checklist from step ten of the AGT-10 instructions to reload the digital computer program. If GATED cannot be executed, obtain the assistance of a laboratory technician. The program tape and data cards must be reloaded following a CRT graphics display failure.

D. DIDITAL COMPUTER PROBLEMS

Improper functioning of the XDS-9300 digital computer can be caused by many problems. The most common problems are listed along with the proper corrections in Ref. 6. Failure to have the peripheral systems in the ready condition usually causes an error message to be printed on the XDS-9300 TTY.

Simply readying the indicated component will, in most cases, clear the system for proper operation.

1. SLOW POINTER MOVEMENT ON CRT DISPLAY

If the altitude pointer on the CRT display moves too slowly or not at all, the XDS-clock switch is probably in the wrong position. Check to see that the clock switch is in the "EXT" position.

2. CARD READER STOPS AFTER FIRST CARD

Check to see that the first card in the deck is a "BOOT" card. If not, place a BOOT card on top and reload the deck. If the BOOT card is in place and the program will still not execute properly, depress IDLE. Then press the CLEAR and CLEAR FLAGS switches simultaneously. This procedure should clear the memory.

Program failure could also occur if the Real Time Monitor (RTM) has been altered by a previous operator. To reload the RTM, execute the following procedures:

a. Mount a "SYSLOAD" tape on a magnetic tape drive. This tape, labeled "RAD DUMP 7/26/74," is located in the tape rack on the left side of the XDS-9300 control console.

b. Set the Unit Select switch to "Ø".

c. Advance the tape to the LOAD POINT.

d. Set the Mode select switch to AUTOMATIC. Set the Mode Select switch on the other tape drive to MANUAL.

e. Press HALT, RESET, RUN, TAPE on the control console. The RTM should now be loaded correctly. Rewind and remove the SYSLOAD tape, and proceed with the simulation operating instructions.

E. SIMULATOR COCKPIT PROBLEMS

1. POWER DOES NOT COME ON

The main power switch is located in the rear of the cockpit. If activiation of this switch fails to switch power to the cockpit, check the circuit breaker box on the wall to the right of the cockpit. Also, check circuit breaker number five, panel R5B, in Room 519.

2. COLLECTIVE MECHANISM MISADJUSTED

Rotation of the collective should cause a voltage change between +30 volts in the upmost position and -23 volts in the lowest position. The collective mechanism can be misaligned by applying too much torque in either direction. An Allen wrench will be needed to realign the gear teeth for full range. The TV camera can be positioned in front of the CI-5000 display console to monitor the "RATIOMETER" for recalibration from the cockpit. Address T000 should be selected to monitor the trunkline which carries the collective voltage signal.

VII. CONCLUSIONS

As stated in Ref. 5, the simulation, as presently configured, becomes unstable as the helicopter reaches the hover position. Divergent pitch and roll oscillations make the helicopter laterally and directionally unstable. Several conditions could cause this instability due to pilot induced oscillations.

Cycle time of the dynamic hybrid loop may be too long, resulting in pilot control inputs lagging display output commands. This problem could possibly be alleviated by "feeding back" cyclic and collective rates to the stability augmentation inputs in the analog program. Investigation of the equations of motion, using a state variable analysis in the hover mode, could also yield some insight into the lateral-directional instability for the hovering conditions.

The arrangement of the aircraft sensor information can be altered by changing the graphics and text portions of the digital program. The present arrangement offers some continuity with the available sensor information. However, the increased availability of multipurpose computers in the future could improve the operational helicopter capabilities. For example, once the helicopter reaches the hover position, a "hover mode" selection switch could change the display for optimum hover control. Upon resumption of normal flight

conditions, the display mode could be switched back to a configuration to accommodate the navigational and control requirements of forward flight.

APPENDIX A

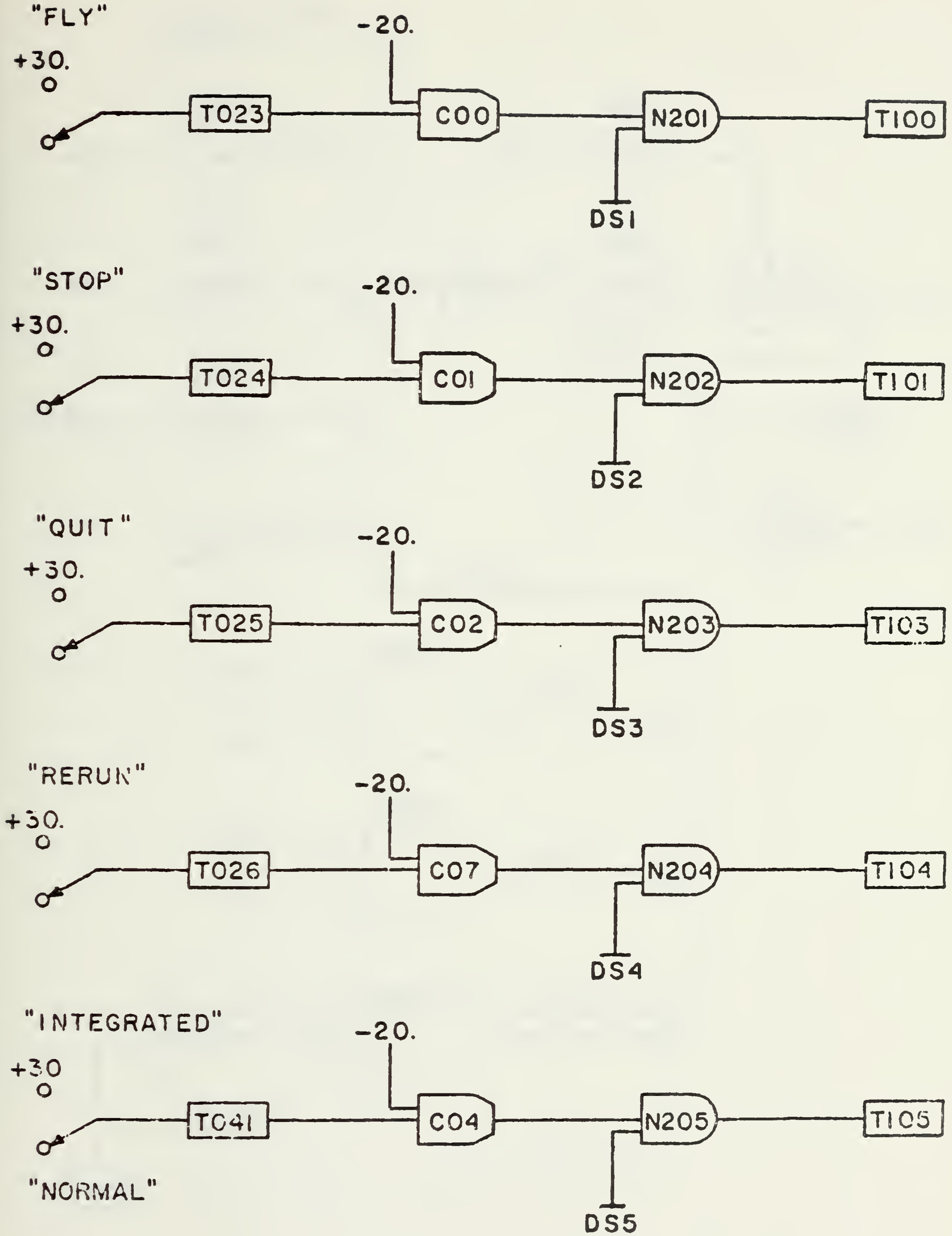
ANALOG COMPUTER PROGRAM

T000	$200\Delta\theta_c$
T001	$200\Delta A_{1c}$
T002	$100\Delta B_{1c}$
T003	+30 VDC
T004	-30 VDC
T005	AIRSPEED
T006	BALL
T007	TURN NEEDLE
T010	HEADING
T011	ALTIMETER
T014	RADAR ALTIMETER
T015	PITCH ATTITUDE
T016	ROLL ATTITUDE
T017	VERTICAL SPEED
T020	$500\Delta\theta_R$
T023	"FLY"
T024	"STOP"
T025	"QUIT"
T026	"RERUN"
T030	COORDINATED TURN
T041	"DISPLAY TYPE"

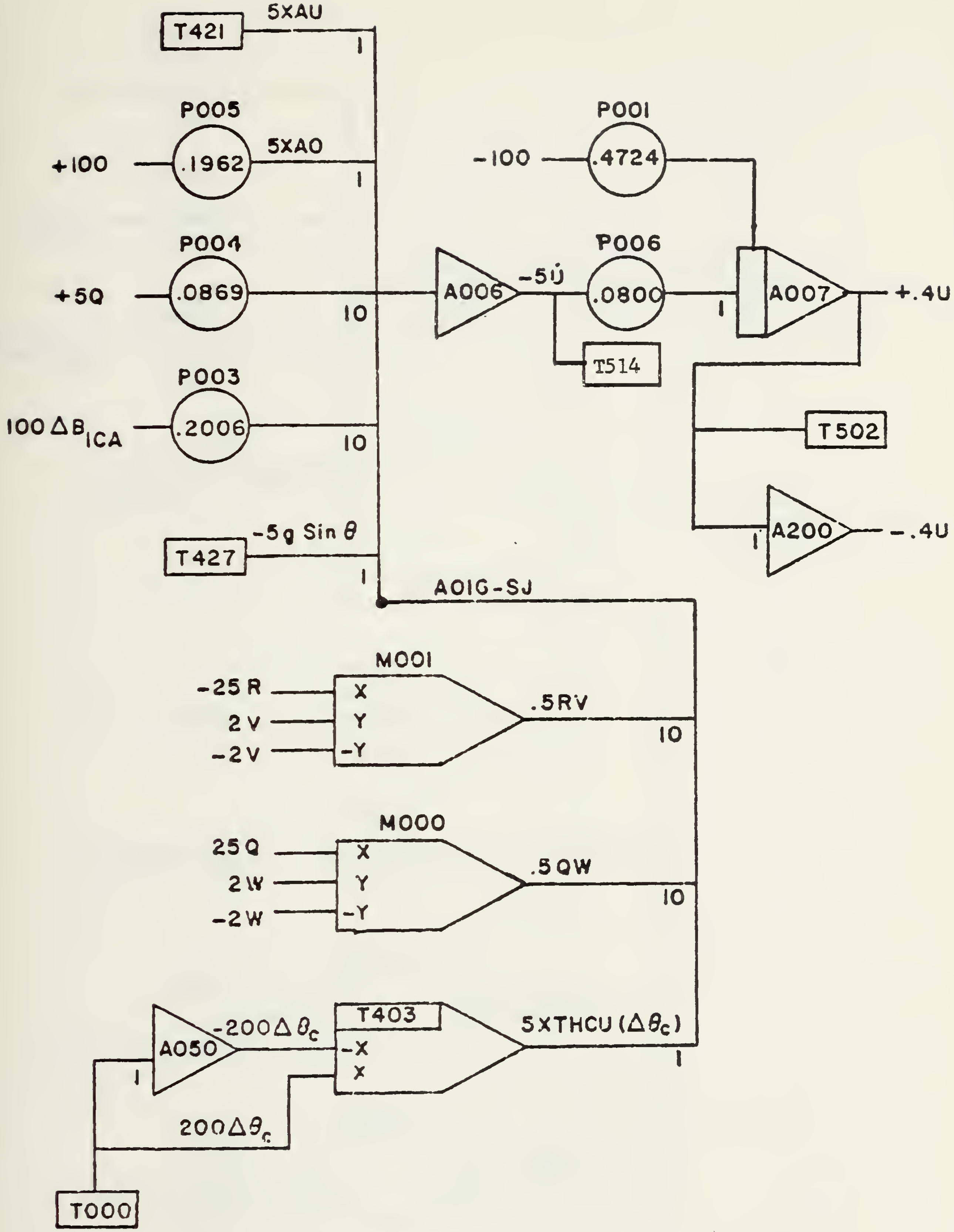
TABLE AI. -- USE OF TRUNK LINES

POT NUMBER	SETTING	POT NUMBER	SETTING
000	.0890	032	.9127
001	.4724	033	.1021
002	.1320	034	.6062
003	.2006	035	.2598
004	.0869	036	.4000
005	.1962	037	.2500
006	.0800	040	.1039
010	.0200	041	.0018
011	.1224	042	.4692
012	.2500	043	.1186
013	.1250	044	.4000
014	.0100	045	.1250
015	.0838	046	.6250
016	.0250	047	.2500
017	.7075	050	.3200
020	.7853	051	.1041
021	.0389	052	.2000
022	.1066	053	.1046
023	.0241	054	.1052
024	.2000	055	.1258
025	.5000	056	.5000
026	.0285	400	.2000
027	.0345	401	.2000
030	.4000	436	.3000
031	.1816	437	.3000

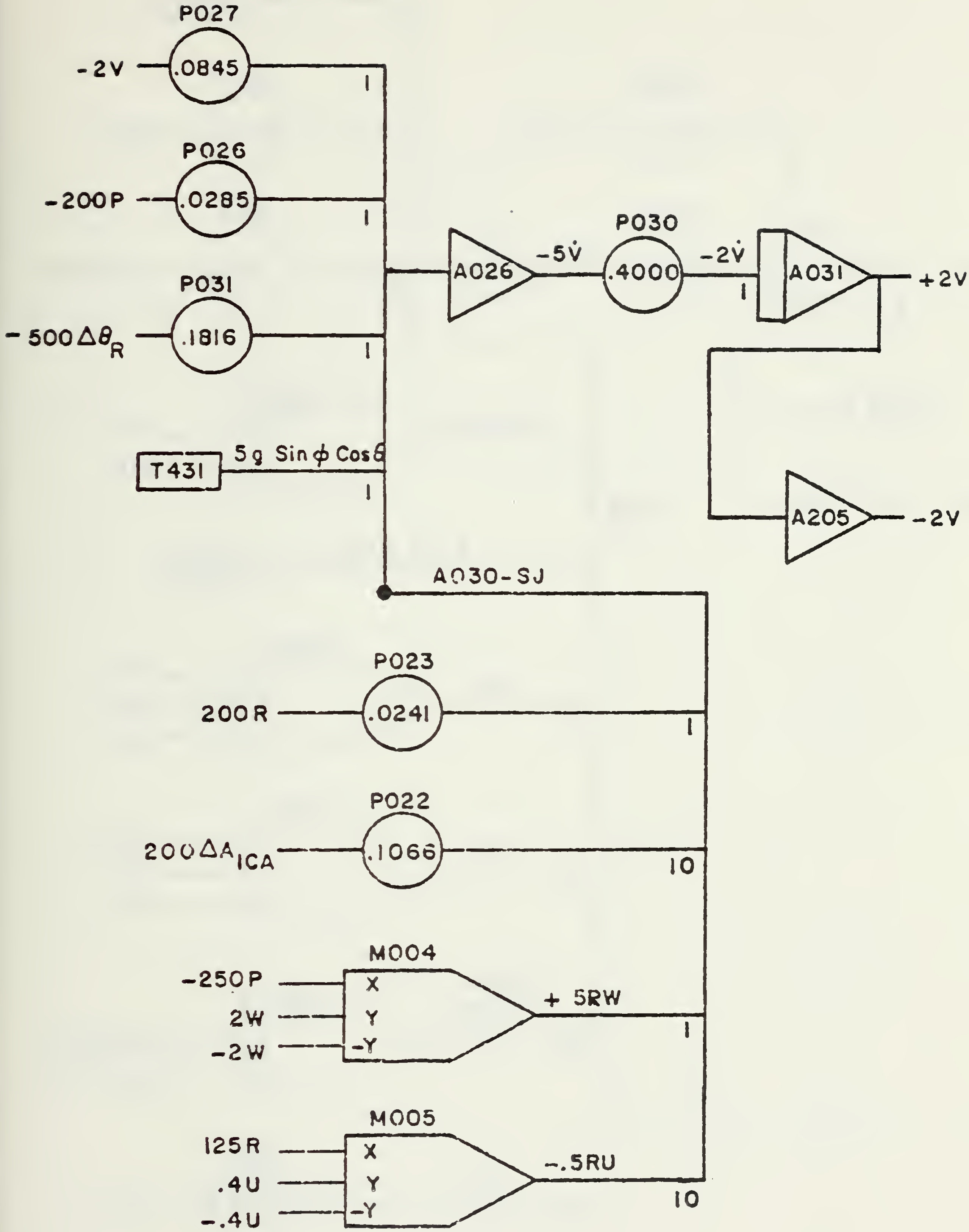
TABLE AII - ANALOG COMPUTER POTENTIOMETER SETTINGS



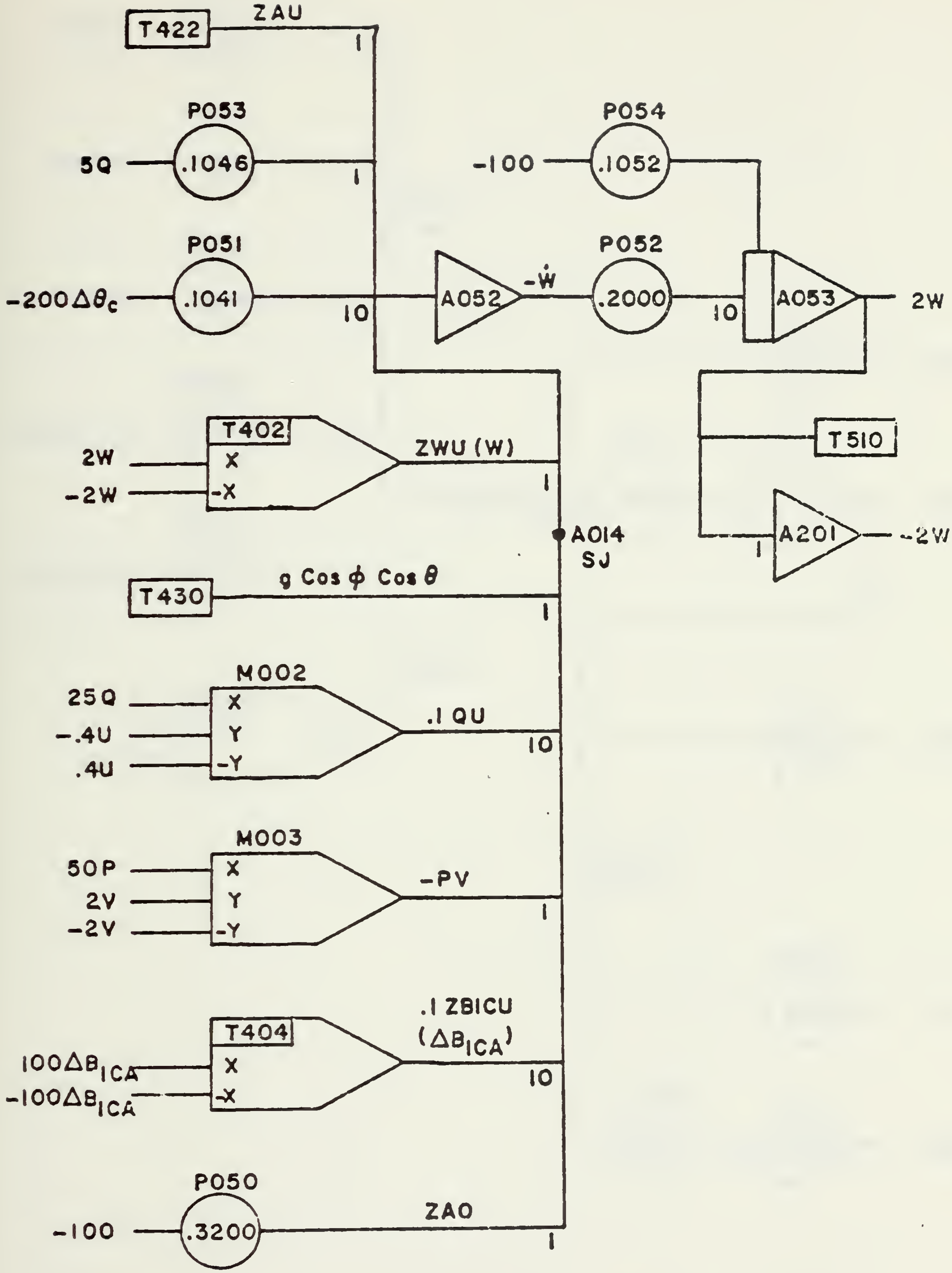
SIMULATION CONTROL



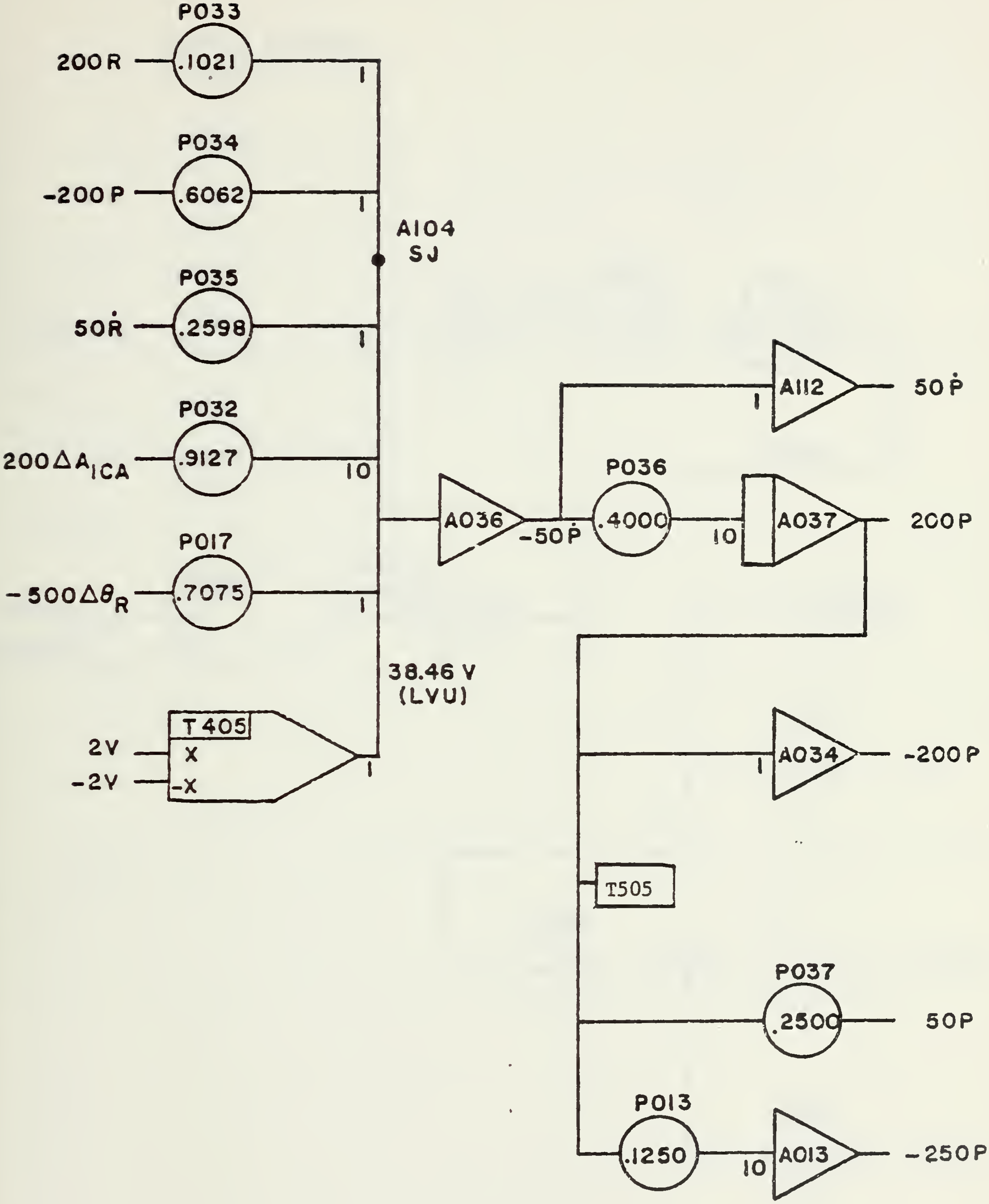
X FORCE EQUATION



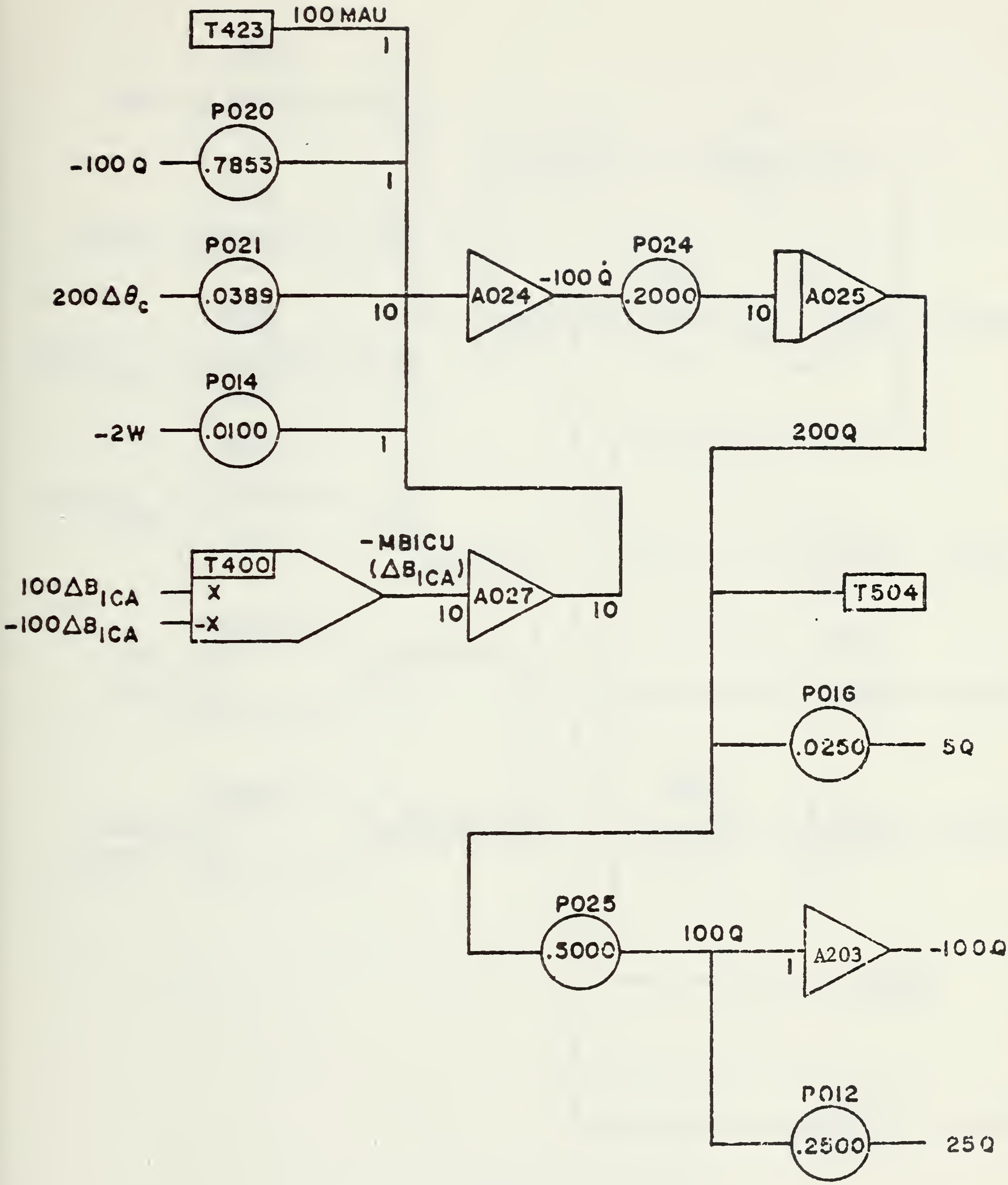
Y FORCE EQUATION



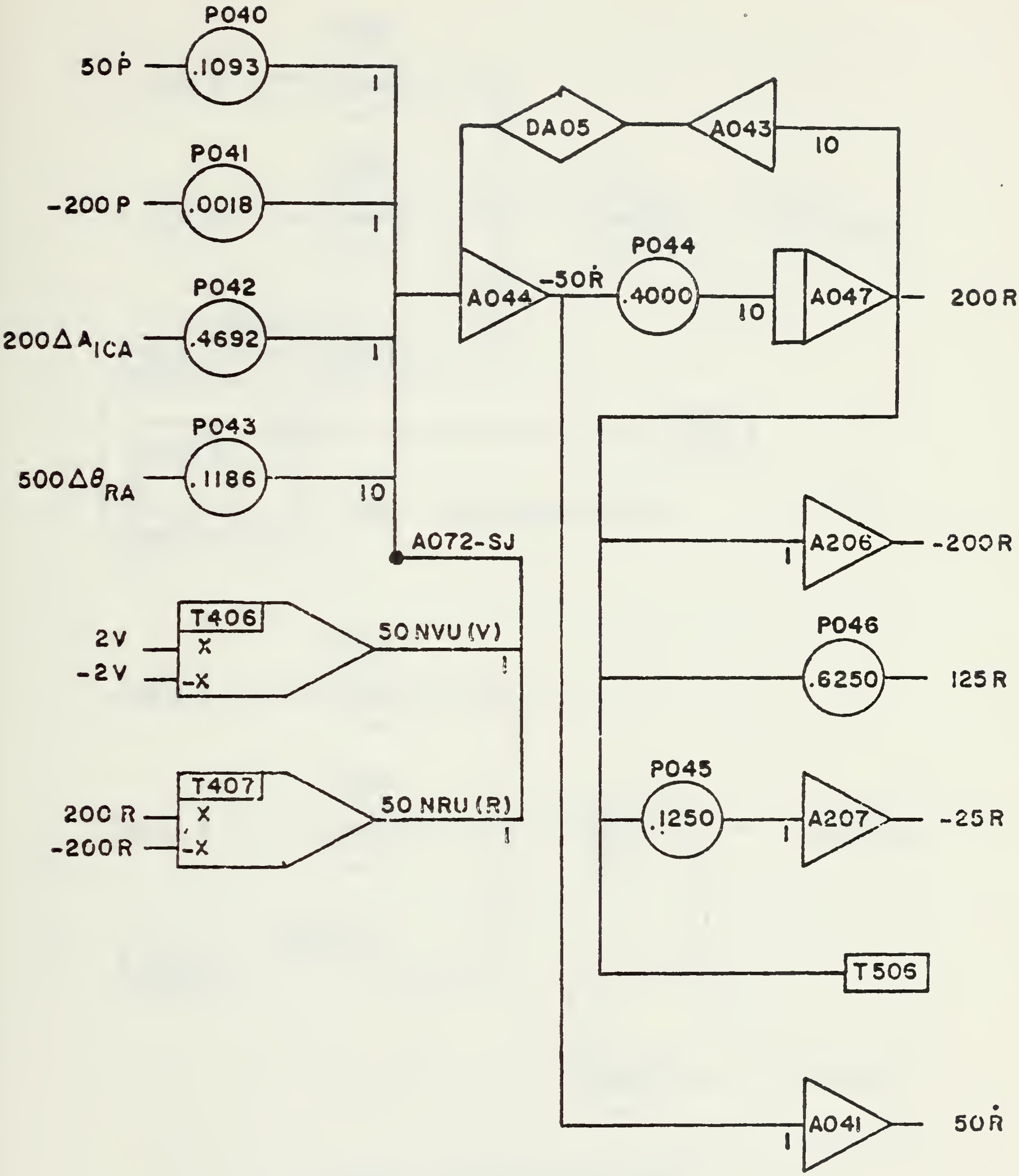
Z FORCE EQUATION



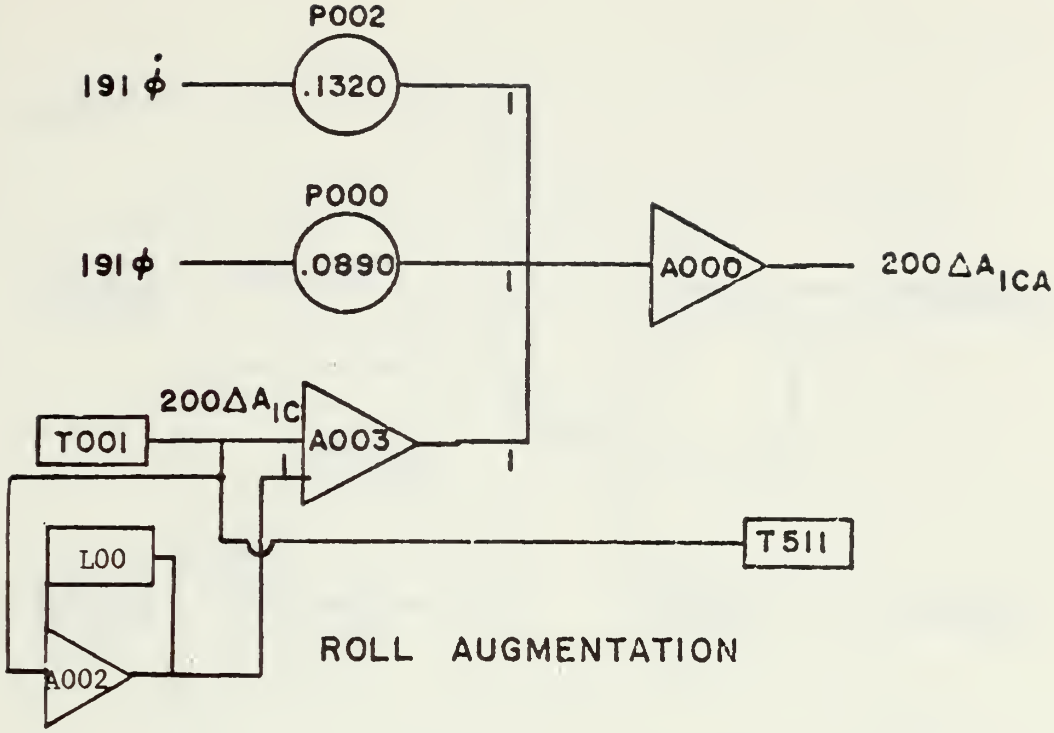
ROLL MOMENT EQUATION



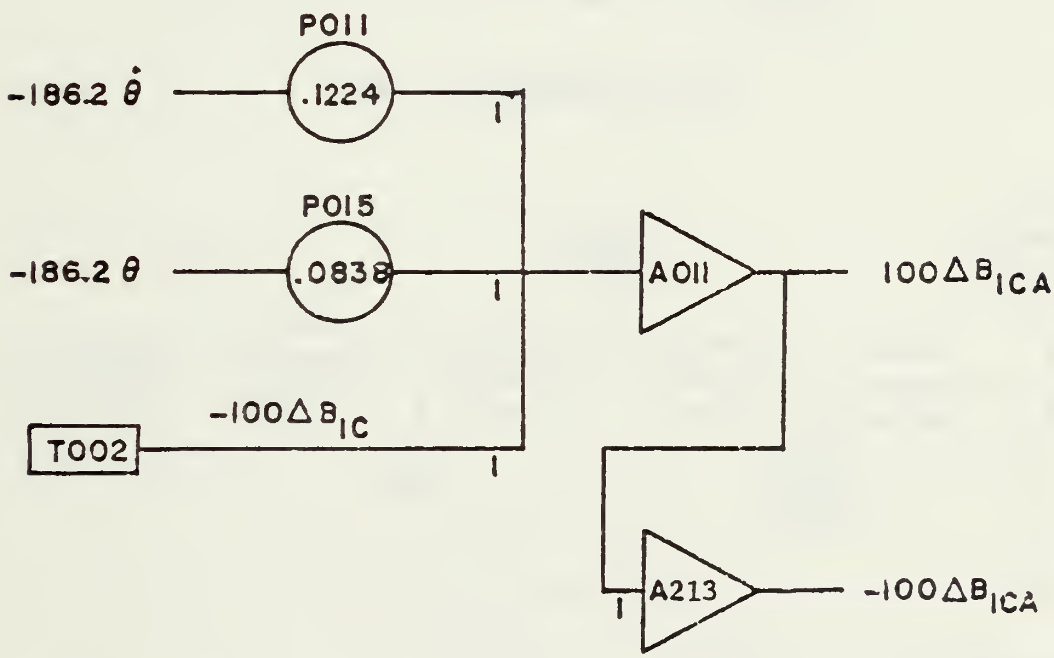
PITCH MOMENT EQUATION



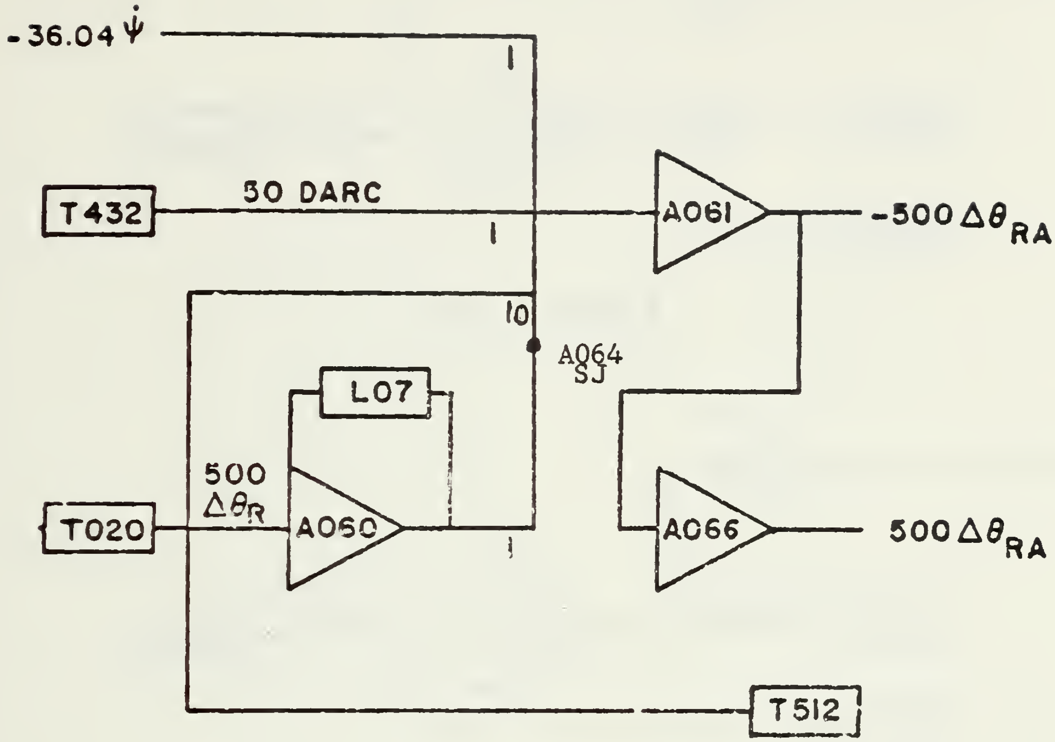
YAW MOMENT EQUATION



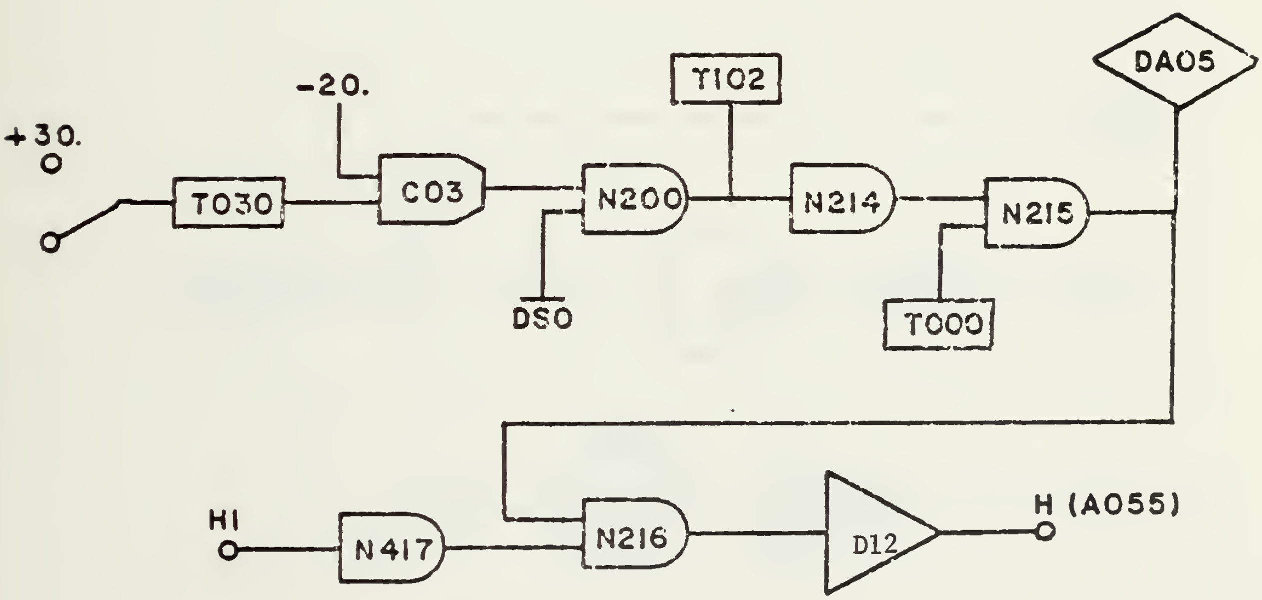
ROLL AUGMENTATION



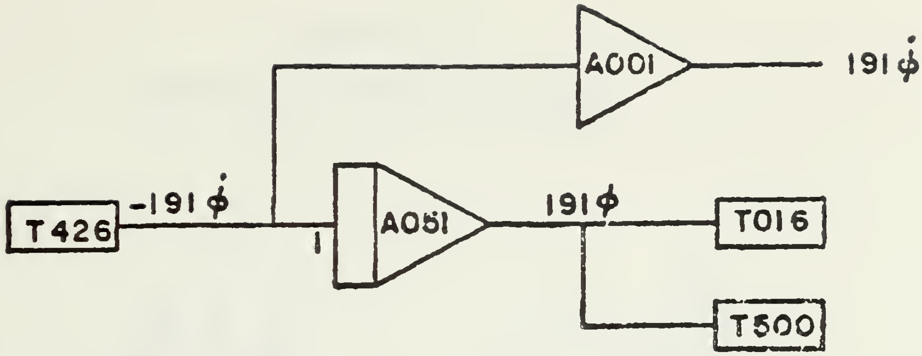
PITCH AUGMENTATION



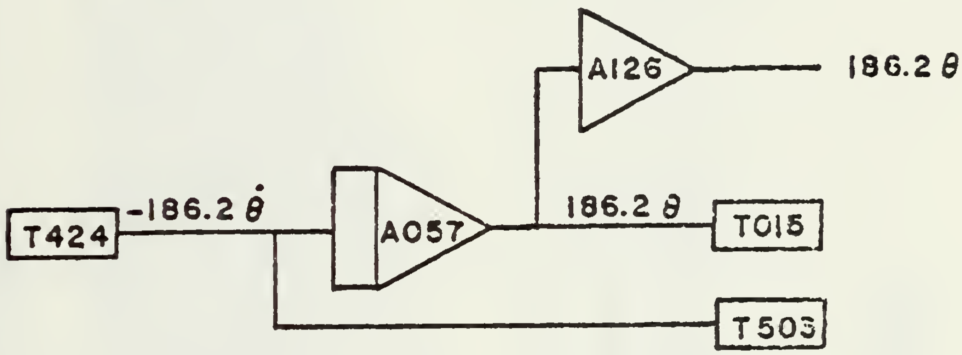
YAW AUGMENTATION



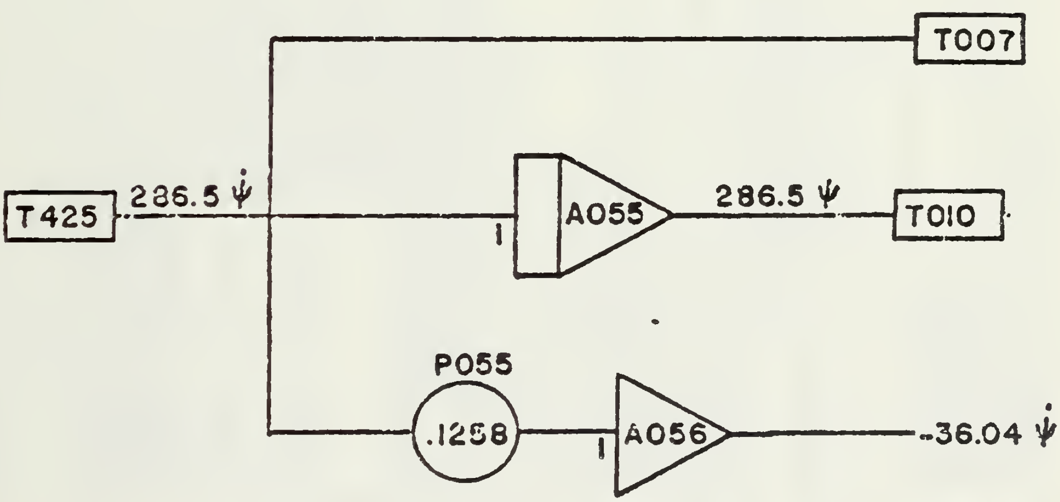
COORDINATED TURN



ROLL ANGLE



PITCH ANGLE



YAW ANGLE

APPENDIX B

DIGITAL COMPUTER PROGRAM

```
***************************************************************************
*                                                                         *
*  EVALUATION OF AN INTEGRATED ELECTRONIC INSTRUMENT DISPLAY FOR          *
*                                                                         *
*    HELICOPTER HOVER OPERATIONS USING A SIX-DEGREE OF FREEDOM            *
*                                                                         *
*          FIXED-BASE SIMULATION OF THE SH-2F HELICOPTER                  *
*                                                                         *
***************************************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      DIMENSION DRVNAM(20)
      COMMON DRVTAB(10,7),CRWDIR(7,17),B
      CCMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      CCMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      CCMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DATA NULL/-1/,ALTNUM/18*-1/,AIRNUM/20*-1/,CMPNUM/90*-1/,
     1     IBLANK/2*0/,HGT/.07/,BASE/.06/,SCALE/.20/,ASCALE/.40/,
     2     DCMPS/.05/,YCHDG/.87/,INDYM(1)/3777B/,INDYM(83)/0/
      NAMELIST IDEV,U,V,W,PHI,THETA,PSI,XE,YE,ZE,UKTS,B
*
*  READ STABILITY DERIVATIVES - PRINT TABULAR LISTING
*
      READ(5,100) DRVNAM
  100 FCRMAT(20A4)
      READ(5,101) ((DRVTAB(I,J),J=1,7),I=1,10)
  101 FCRMAT(7F10.4)
      WRITE(6,102)
  102 FORMAT('1',/////,40X,'STABILITY DERIVATIVES'//)
      WRITE(6,103)
  103 FORMAT('0',20X,'0 KTS',5X,'30 KTS',5X,'50 KTS',5X,'70 KTS',5X,
     1       '91 KTS',4X,'112 KTS',4X,'136 KTS',/)
      WRITE(6,104) (DRVNAM(2*I-1),DRVNAM(2*I),(DRVTAB(I,J),J=1,7),I=1,10)
  104 FORMAT('0',8X,2A4,F9.4,6F11.4)
*
*  READ CREW DIRECTIONS - PRINT LISTING
*
      READ(5,105) ((CRWDIR(I,J),I=1,6),J=1,17)
  105 FCRMAT(6A4)
      WRITE(6,106)
  106 FORMAT('1',/////,41X,'CREW DIRECTIONS'/)
      WRITE(6,107) ((CRWDIR(I,J),I=1,6),J=1,17)
  107 FORMAT('0',17X,18A4)
```

```
*
*   READ RADAR ALTIMETER SCALE NUMBERS - ECHO CHECK
*
      READ(5,108) (ALTNUM(J),J=1,17,2)
  108 FCRMAT(20A4)
      WRITE(6,109)
  109 FORMAT(//'0',17X,'ALTITUDE SCALE NUMBERS')
      WRITE(6,110) (ALTNUM(J),J=1,17,2)
  110 FORMAT('0',14X,20A4//)
*
*   READ AIRSPEED SCALE NUMBERS - ECHO CHECK
*
      READ(5,108) (AIRNUM(J),J=1,19,2)
      WRITE(6,111)
  111 FCRMAT('0',17X,'AIRSPEED SCALE NUMBERS')
      WRITE(6,112) (AIRNUM(J),J=1,19,2)
  112 FCRMAT('0',17X,20A4//)
*
*   READ COMPASS NUMBERS - ECHO CHECK
*
      READ(5,113) (CMPNUM(J),J=1,90)
  113 FCRMAT(48A1)
      WRITE(6,114)
  114 FORMAT('0',17X,'COMPASS NUMBERS')
      WRITE(6,115) (CMPNUM(J),J=1,90)
  115 FCRMAT('0',16X,48A1,/,' ',16X,48A1)
*
*   SET POTENTIOMETERS
*
      CALL SETPOT (4HP000,0890, 4HP001,4724, 4HP002,1320, 4HP003,2006,
     1             4HP004,0869, 4HP005,1962, 4HP006,0800, 4HP007,0000,
     2             4HP010,0200, 4HP011,1224, 4HP012,2500, 4HP013,1250,
     3             4HP014,0100, 4HP015,0838, 4HP016,0250, 4HP017,7075,
     4             4HP020,7853, 4HP021,0389, 4HP022,1066, 4HP023,0241,
     5             4HP024,2000, 4HP025,5000, 4HP026,0285, 4HP027,0845,
     6             4HP030,4000, 4HP031,1816, 4HP032,9127, 4HP033,1021,
     7             4HP034,6062, 4HP035,2598, 4HP036,4000, 4HP037,2500,
     8             4HP040,1039, 4HP041,0018, 4HP042,4692, 4HP043,1186,
     9             4HP044,4000, 4HP045,1250, 4HP046,6250, 4HP047,2500,
     A             4HP050,3200, 4HP051,1041, 4HP052,2000, 4HP053,1046,
     B             4HP054,1052, 4HP055,1258, 4HP056,5000, 4HP057,1128)
*
*   SELECT GRAPHICS COMPUTER
*
      OUTPUT(101) 'SELECT GRAPHICS COMPUTER - TYPE: IDEV=  *'
      INPUT(101)
*
*   SET INITIAL FLIGHT CONDITIONS
```

```
*
 200 CALL RESET(1000)
     UKTS=70.
     U=UKTS*1.687
     V=0.
     W=5.26
     P=0.
     Q=0.
     R=0.
     PHI=0.
     THETA=2.55/57.3
     PSI=0.
     PHIDOT=0.
     THEDOT=0.
     PSIDOT=0.
     XE=-2.*2000.
     YE=0.
     ZE=500.
     B=.3
*
*  INPUT ALTERNATE INITIAL CONDITIONS IF DESIRED
*
     IF(SENSE SWITCH 3) 202,205
 202 OUTPUT(101) 'INPUT ALTERNATE INITIAL CONDITIONS'
     INPUT(101)
*
*  INITIALIZE GRAPHICS COMPUTER
*
 205 NTD=20
     NGD=20
     CALL DTINIT(IDEV,ITD,NTD,IER)
     IF(IER.NE.0) OUTPUT(6) 'ERROR - DTINIT',IER
     CALL DGINIT(IDEV,IGD,NGD,IER)
     IF(IER.NE.0) OUTPUT(6) 'ERROR - DGINIT',IER
*
*  GENERATE STATIC PORTION OF INSTRUMENT DISPLAY
*
     CALL DSPLY
*
*  COMMENCE DYNAMIC SIMULATION
*
     CALL FLY
*
*  COMPUTE RMS INERTIAL VELOCITIES AND RMS POSITIONS
*
     VXRMS=SQRT(VXRMS/RMSTIM)
     VYRMS=SQRT(VYRMS/RMSTIM)
     VZRMS=SQRT(VZRMS/RMSTIM)
```

```
      XERMS=SQRT(XERMS/RMSTIM)
      YERMS=SQRT(YERMS/RMSTIM)
      ZERMS=SQRT(ZERMS/RMSTIM)
      OUTPUT(6) ' ',VXRMS,VYRMS,VZRMS,XERMS,YERMS,ZERMS,RMSTIM,FLTIM
*
*  CHECK FOR QUIT OR RERUN
*
  210 IF(TEST(5).LT.0)  GO TO 200
      IF(TEST(4).LT.0)  GO TO 220
      GO TO 210
  220 CALL POTSET
      STOP
      END
```

```
      SUBROUTINE DSPLY
*****************************************************
*                                                   *
*   GENERATES STATIC PORTION OF INSTRUMENT DISPLAY  *
*                                                   *
*****************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      CCMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      CCMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSION IVSI(20),IDVI(38),IALT(59),IVGI(20),IMAP(9),ITURN(17),
     1          ISLIP(6),ISPD(38)
      NBLK=1
*                                                                      KAR00680
*   VERTICAL SPEED INDICATOR - SCALE DIVISIONS
*
      XOVSI=-2.50*SCALE
      YOVSI=-1.95*SCALE
      DS=.10*SCALE
      DL=.22*SCALE
      DVSI=.35*SCALE
      Y=YOVSI-4.*DVSI
      IVSI(1)=IHEAD(0,10)
      IVSI(2)=IPACK(XOVSI,Y,0)
      DO 110 J=3,18,4
        X=XOVSI+DL
        IVSI(J)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DVSI
        IVSI(J+1)=IPACK(X,Y,0)
        X=X+DS
        IVSI(J+2)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DVSI
        IVSI(J+3)=IPACK(X,Y,0)
  110   CONTINUE
      X=X+DL
      IVSI(19)=IPACK(X,Y,1)
      IVSI(20)=0
      CALL GRAPHO(IDEV,IVSI,20,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IVSI',IER,*
      XPVSI=X+.10*SCALE
*
*   DIRECTION VELOCITY INDICATOR - SCALE DIVISIONS
```

```
*
      XODVI=0.
      YODVI=YOVSI
      DI=0.20*SCALE
      DDVI=0.35*SCALE
      X=XODVI-4.*DDVI
      Y=YODVI-DL/2.
      IDVI(1)=IHEAD(0,10)
      IDVI(2)=IPACK(X,Y,0)
      DO 115 J=3,19,2
        Y=Y+DL
        IDVI(J)=IPACK(X,Y,1)
        X=X+DDVI
        Y=Y-DL
        IDVI(J+1)=IPACK(X,Y,0)
  115   CONTINUE
      X=XODVI-DL/2.
      Y=YODVI-4.*DDVI
      IDVI(20)=IPACK(X,Y,0)
      DO 120 J=21,37,2
        X=X+DL
        IDVI(J)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DDVI
        IDVI(J+1)=IPACK(X,Y,0)
  120   CONTINUE
      IDVI(38)=0
      CALL GRAPHO(IDEV,IDVI,38,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IDVI',IER,*
      XLEFT= XODVI-4.*DDVI
      XRIGHT=XODVI+4.*DDVI
      YTOP=YODVI+4.0*DDVI
      YBOT=YODVI-4.0*DDVI
*
*   CHECK FOR DESIRED DISPLAY
*
      IF(TEST(6).GE.0) RETURN
*
*   RADAR ALTIMETER - SCALE NUMBERS                                 KAR00690
*                                                                   KAR00700
      LN0=36
      LN=LN0
      ICOPS=8
      DO 108 J=1,11,2                                               KAR00790
        CALL TEXTO(IDEV,ALTNUM(J),2,LN,ICOPS,2,3,IER)
        IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBERS',J,IER,* KAR00810
        LN=LN-4                                                     KAR00820
```

```
108   CONTINUE                                                          KAR00830
      LN=LN+2                                                           KAR00840
      CALL TEXTO(IDEV,ALTNUM(13),2,LN,ICOPS,2,3,IER)
      IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBER 7',IER,*
      LN=LN-2
      CALL TEXTO(IDEV,ALTNUM(15),2,LN,ICOPS,2,3,IER)
      IF(IER.NE.0) OUTPUT(6),'ERROR--ALTIMETER SCALE NUMBER 8',IER,*    KAR00890
      LN=LN-3
      CALL TEXTO(IDEV,ALTNUM(17),2,LN,ICOPS,2,3,IER)
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ALTIMETER SCALE NUMBER 9',IER,*
*
*  RADAR ALTIMETER - SCALE
*
      XOALT=.125*(ICOPS-40.)*SCALE
      YOALT=.3*(21.5-LNO)*SCALE
      DS=.10*SCALE
      DL=.25*SCALE
      DALT=.30*SCALE
      Y=YOALT
      IALT(1)=IHEAD(0,10)
      IALT(2)=IPACK(XOALT,YOALT,0)
      DO 102 J=3,51,4
        X=XOALT+DL
        IALT(J)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DALT
        IALT(J+1)=IPACK(X,Y,0)
        X=X+DS
        IALT(J+2)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DALT
        IALT(J+3)=IPACK(X,Y,0)
102   CONTINUE
      X=X+DL-DS
      IALT(55)=IPACK(X,Y,0)
      X=X+DS
      IALT(56)=IPACK(X,Y,1)
      X=X-DL
      Y=Y+DALT
      IALT(57)=IPACK(X,Y,0)
      X=X+DL
      IALT(58)=IPACK(X,Y,1)
      IALT(59)=0
      CALL GRAPHO(IDEV,IALT,59,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IALT',IER,*
      XPALT=X+.10*SCALE
*
```

```
*   VERTICAL GYRO INDICATOR - ANGLE OF BANK SCALE                  KAR01550
*                                                                  KAR01560
      XCEN=0.                                                      KAR01570
      YCEN= 1.5*SCALE
      R=2.0*SCALE                                                  KAR01590
      IVGI(1)=IHEAD(0,10)
      J=2
      REPEAT 122, FOR ROLL=-60.,-30.,-20.,-10.,0.,10.,20.,30.,60.
        RL=2.35*SCALE                                              KAR01620
        AROLL=ABS(ROLL)
        IF(AROLL.EQ.10..OR.AROLL.EQ.20.) RL=2.20*SCALE
        Y=YCEN+R*COS(ROLL/57.3)
        X=XCEN+R*SIN(ROLL/57.3)
        IVGI(J)=IPACK(X,Y,0)                                       KAR01660
        X=XCEN+RL*SIN(ROLL/57.3)
        Y=YCEN+RL*COS(ROLL/57.3)
        IVGI(J+1)=IPACK(X,Y,1)                                     KAR01690
        J=J+2
  122   CONTINUE                                                   KAR01710
      IVGI(20)=0                                                   KAR01720
      CALL GRAPHO(IDEV,IVGI,20,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IVGI',IER,*                   KAR01740
      RP=R-.05*SCALE
      RB=RP-HGT
      EPS=ATAN(BASE/(2.*RB))
*                                                                  KAR01750
*   VERTICAL GYRO INDICATOR - MINIATURE AIRPLANE                   KAR01760
*                                                                  KAR01770
      DL=1.0*SCALE                                                 KAR01780
      DS=0.2*SCALE                                                 KAR01790
      IMAP(1)=IHEAD(0,10)                                          KAR01800
      X=XCEN-DL                                                    KAR01810
      IMAP(2)=IPACK(X,YCEN,0)                                      KAR01820
      X=XCEN-2.*DS                                                 KAR01830
      IMAP(3)=IPACK(X,YCEN,1)                                      KAR01840
      X=XCEN-DS                                                    KAR01850
      Y=YCEN-DS                                                    KAR01860
      IMAP(4)=IPACK(X,Y,1)                                         KAR01870
      IMAP(5)=IPACK(XCEN,YCEN,1)                                   KAR01880
      IMAP(6)=IPACK(-X,Y,1)                                        KAR01890
      X=XCEN+2.*DS                                                 KAR01900
      IMAP(7)=IPACK(X,YCEN,1)                                      KAR01910
      X=XCEN+DL                                                    KAR01920
      IMAP(8)=IPACK(X,YCEN,1)                                      KAR01930
      IMAP(9)=0                                                    KAR01940
      CALL GRAPHO(IDEV,IMAP,9,NBLK,IER)
      NBLK=NBLK+1
```

```
      IF(IER.NE.0) OUTPUT(6) 'ERROR--IMAP',IER,*                      KAR01960
*
*  SLIP INDICATOR - SCALE
*
      XOSLIP= 3.60*SCALE
      YOSLIP= 2.10*SCALE
      DS= .25*SCALE
      X=XOSLIP-DS/2.
      Y=YOSLIP+DS/2.
      ISLIP(1)=IHEAD(0,10)
      ISLIP(2)=IPACK(X,Y,0)
      Y=Y-DS
      ISLIP(3)=IPACK(X,Y,1)
      X=X+DS
      ISLIP(4)=IPACK(X,Y,0)
      Y=Y+DS
      ISLIP(5)=IPACK(X,Y,1)
      ISLIP(6)=0
      CALL GRAPHO(IDEV,ISLIP,6,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ISLIP',IER,*
*                                                                     KAR02290
*  RATE OF TURN - SCALE
*
      XOTURN=XOSLIP
      YOTURN=YOSLIP+0.90*SCALE
      DS=.10*SCALE
      DL=.20*SCALE
      DTURN=.70*SCALE
      ITURN(1)=IHEAD(0,10)
      X=XOTURN-DTURN+DL/2.
      Y=YOTURN
      ITURN(2)=IPACK(X,Y,0)
      DO 135 J=3,13,5
        X=X-DL
        ITURN(J)=IPACK(X,Y,1)
        Y=Y+DS
        ITURN(J+1)=IPACK(X,Y,1)
        X=X+DL
        ITURN(J+2)=IPACK(X,Y,1)
        Y=Y-DS
        ITURN(J+3)=IPACK(X,Y,1)
        X=X+DTURN
        ITURN(J+4)=IPACK(X,Y,0)
 135    CONTINUE
      ITURN(17)=0
      CALL GRAPHO(IDEV,ITURN,17,NBLK,IER)
      NBLK=NBLK+1
```

```
      IF(IER.NE.0) OUTPUT(6)  'ERROR--ITURN',IER,*
*
*  AIRSPEED - SCALE NUMBERS                                            KAR02300
*                                                                      KAR02310
      LNO=36
      LN=LNO
      ICOPS=80
      DO 130 J=1,20,2                                                  KAR02380
        CALL TEXTO(IDEV,AIRNUM(J),2,LN,ICOPS,2,3,IER)
        IF(IER.NE.0) OUTPUT(6),'ERROR--AIRPSEED SCALE NUMBERS',J,IER,*  KAR02400
        LN=LN-2                                                        KAR02410
 130    CONTINUE                                                       KAR02420
*
*  AIRSPEED - SCALE DIVISIONS
*
      XOSPD=.125*(ICOPS-52.)*SCALE
      YOSPD=.3*(21.5-LNO)*SCALE
      DS=.10*SCALE
      DL=.25*SCALE
      DSPD=.30*SCALE
      ISPD(1)=IHEAD(0,10)
      ISPD(2)=IPACK(XOSPD,YOSPD,0)
      X=XOSPD+DL
      ISPD(3)=IPACK(X,YOSPD,1)
      X=X-DL
      Y=YOSPD+2.*DSPD
      ISPD(4)=IPACK(X,Y,0)
      DO 125 J=5,33,4
        X=X+DL
        ISPD(J)=IPACK(X,Y,1)
        X=X-DL
        Y=Y+DSPD
        ISPD(J+1)=IPACK(X,Y,0)
        X=X+DS
        ISPD(J+2)=IPACK(X,Y,1)
        X=X-DS
        Y=Y+DSPD
        ISPD(J+3)=IPACK(X,Y,0)
 125    CONTINUE
      X=X+DL
      ISPD(37)=IPACK(X,Y,1)
      ISPD(38)=0
      CALL GRAPHO(IDEV,ISPD,38,NBLK,IER)
      NBLK=NBLK+1
      IF(IER.NE.0) OUTPUT(6) 'ERROR--ISPD',IER,*
      XPSPD=X-DL-.10*SCALE
      RETURN
      END
```

```
      SUBROUTINE FLY
*****************************************************
*                                                   *
*  DYNAMIC SIMULATION                               *
*                                                   *
*****************************************************
      REAL MAU,MB1CU,LVU,NRU,NVU
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      COMMON DRVTAB(10,7),CRWDIR(7,17),B
      CCMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      COMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      CCMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSION DRV(10)
      EQUIVALENCE (XAU,DRV(1)),(ZAU,DRV(2)),(MAU,DRV(3)),(ZWU,DRV(4)),
     1            (MB1CU,DRV(5)),(LVU,DRV(6)),(NRU,DRV(7)),(NVU,DRV(8)),
     2            (XTHCU,DRV(9)),(ZB1CU,DRV(10))
*
*  SET INITIAL CONDITIONS - SAVE INITIAL VALUES
*
      CALL WRITECLOCK(0)
      IFLAG=0
      JFLAG=0
      KFLAG=0
      FLTIM=0.
      RMSTIM=0.
      VXRMS=0.
      VYRMS=0.
      VZRMS=0.
      XERMS=0.
      YERMS=0.
      ZERMS=0.
      TOLD=0.
      THETIC=THETA
      DA1C=0.
      DTHR=0.
      KSAV=0
*
  100.IF(IFLAG.EQ.1) GO TO 110
      GC TO 115
  110 CALL COMPUTE
      CALL STARTCLOCK
*
```

```
*  PRINT AIRCRAFT STATE VARIABLES IF DESIRED
*
 115 IF(SENSE SWITCH 1) 120,130
 120 WRITE(6,125) UKTS,V,W,P,Q,R,PHI,THETA,PSI,XE,YE,ZE,FLTIM
 125 FORMAT('0',F5.1,12F9.2)
*
*  CALCULATE VALUES OF AIRSPEED DEPENDENT STABILITY DERIVATIVES
*
 130 DO 137 I=1,10
       IF(UKTS.LE.0.) DRV(I)=DRVTAB(I,1) ; GO TO 137
       J=1
       U1=0.
       REPEAT 135, FOR U2= 30.,50.,70.,91.,112.,136.
       IF(UKTS.GE.U2) GO TO 132
       DRV(I)=DRVTAB(I,J)+(DRVTAB(I,J+1)-DRVTAB(I,J))*(UKTS-U1)/(U2-U1)
       GO TO 137
 132   J=J+1
       U1=U2
 135   CONTINUE
       DRV(I)=DRVTAB(I,7)
 137   CONTINUE
     IF(SENSE SWITCH 2) 500,502
 500 WRITE(6,501) UKTS,XAU,ZAU,LVU
 501 FORMAT('0',4F12.4)
 502 CONTINUE
*
*  SCALE STABILITY DERIVATIVES
*
     XAU=XAU*.05
     ZAU=ZAU*.01
*    MAU IS OK
     ZWU=ZWU*.5
     MB1CU=-MB1CU*.01
     IVU=LVU*19.23
     NRU=NRU*.25
     NVU=NVU*25.
     XTHCU=XTHCU*.025
     ZB1CU=ZB1CU*.001
*
*  CCMPUTE APPROPIRATE TRIG FUNCTIONS TO SAVE COMPUTER TIME
*
     SINPHI=SIN(PHI)
     CCSPHI=COS(PHI)
     SINTHE=SIN(THETA)
     CCSTHE=COS(THETA)
     SINPSI=SIN(PSI)
     COSPSI=COS(PSI)
*
```

```
*   COMPUTE REQUIRED FUNCTIONS OF EULER ANGLES - SCALE
*
*        -5G SIN(THETA)
      GSINTH=-1.61*SINTHE
*        G COS(PHI) COS(THETA)
      GCOSPH=.322*COSPHI*COSTHE
*         5G SIN(PHI) COS(THETA)
      GSINPH=1.61*SINPHI*COSTHE
*
*   COORDINATED TURN SYSTEM - DETERMINE TAIL ROTOR INPUT TO ANALOG
*
      DA1C=DA1C*.5
*     CHECK COORDINATED TURN SWITCH
      IF(TEST(3).LT.0) GO TO 150
      DTHR=ABS(DTHR)
      IF(DTHR.GT..003) CALL SETLINES(1,1) ; GO TO 140
      CALL SETLINES(1,-1)
  140 DTHRC=0.
      GO TO 155
  150 CALL SETLINES(1,-1)
      BETA=ATAN(V/U)
      DTHRC=16.*DA1C+10.*BETA-3.*PSIDOT
*
*   COMPUTE EULER ANGLE RATES - SCALE FOR ANALOG INPUT
*
  155 THEDOT=Q*COSPHI-R*SINPHI
      PSIDOT=(Q*SINPHI+R*COSPHI)/COSTHE
      PHIDOT=P+PSIDOT*SINTHE
      THEDTS=-THEDOT*1.862
      PSIDTS= PSIDOT*2.865
      PHIDTS=-PHIDOT*1.91
*
*
*   COMPUTE INERTIAL VELOCITIES - SCALE VZ FOR ANALOG INPUT
*
      V1=U*COSTHE+(V*SINPHI+W*COSPHI)*SINTHE
      V2=V*COSPHI-W*SINPHI
      VX=V1*COSPSI-V2*SINPSI
      VY=V1*SINPSI+V2*COSPSI
      VZ=-U*SINTHE+V*SINPHI*COSTHE+W*COSPHI*COSTHE
      VZS=VZ*.05
*
*   AIRSPEED INDICATOR - COMPUTE SIGNAL FOR COCKPIT INSTRUMENT
*
      IF(UKTS.GT.60.) GO TO 160
      ASI=.01*(59.-.4765*UKTS)
      GO TO 180
  160 IF(UKTS.GT.90.) GO TO 170
```

```
      ASI=.01*(100.-(10./9.)*UKTS)
      GO TO 180
  170 ASI=.01*(163.6-(100./55.)*UKTS)
*
*  RADAR ALTIMETER - COMPUTE SIGNAL FOR COCKPIT INSTRUMENT
*
  180 IF(ZE.GT.200.)  RADALT=-.65                          ; GO TO 190
      IF(ZE.GT.100.)  RADALT=-.40-.001983*(ZE-100.)        ; GO TO 190
      IF(ZE.GT.70.)   RADALT=-.345-.00183*(ZE-70.)         ; GO TO 190
      IF(ZE.GT.50.)   RADALT=-.300-.00225*(ZE-50.)         ; GO TO 190
      IF(ZE.GT.40.)   RADALT=-.2893-.00107*(ZE-40.)        ; GO TO 190
      IF(ZE.GT.20.)   RADALT=-.2429-.00232*(ZE-20.)        ; GO TO 190
      IF(ZE.GE.0.)    RADALT=-.2129-.0015*ZE               ; GO TO 190
      IF(ZE.LT.0.)    RADALT=-.2129
*
*  COMPUTE INERTIAL POSITION AND FLIGHT TIME
*
  190 CALL READCLOCK(TNEW)
      DT=.0001*(TNEW-TOLD)
      TOLD=TNEW
*     XE AND YE IN YARDS, ZE IN FEET
      XE=XE+VX*DT/3.
      YE=YE+VY*DT/3.
      ZE=ZE-VZ*DT
      FLTIM=.0001*TNEW
*
*  ACCUMULATE DATA TO COMPUTE RMS INERTIAL VELOCITIES AND POSITIONS
*
      IF(KFLAG.EQ.0) GO TO 195
      RMSTIM=RMSTIM+DT
      VXRMS=VXRMS+VX*VX*DT
      VYRMS=VYRMS+VY*VY*DT
      VZRMS=VZRMS+VZ*VZ*DT
      XERMS=XERMS+XE*XE*DT
      YERMS=YERMS+YE*YE*DT
      ZERMS=ZERMS+ZE*ZE*DT
*
*  INSTRUMENT DISPLAY DYNAMICS
*
  195 CALL INST
*
*  CREW DIRECTIONS
*
      IF(XE.LE.2000..AND.ZE.LE.250.) CALL CREW
*
*  PERFORM D-A AND A-D CONVERSIONS
*
      CALL DAC(10,VZS,2,XAU, 3,ZAU, 4,MAU, 5,THEDTS, 6,PSIDTS, 7,PHIDTS,
```

```
     1          8,GSINTH, 9,GCOSPH, 1, GSINPH, 11,DTHRC, 12,ASI,
     2          16,MB1CU, 17,RADALT, 18,ZWU, 19,XTHCU, 20,ZB1CU, 21,LVU,
     3          22,NVU, 23,NRU)
      CALL ADK(0,PHI, 2,U, 3,THETA, 4,Q, 5,P, 6,R, 7,V, 8,W, 9,DA1C,
     1          10,DTHR)
*
*  SCALE VARIABLES AS REQUIRED
*
      U=U*250.
      UKTS=U/1.687
      V=V*50.
      W=W*50.
      P=P*.5
      Q=Q*.5
      R=R*.5
      PHI=PHI/1.91
      THETA=THETA/1.862+THETIC
      PSI=PSI+PSIDOT*DT
*
*  CHECK FOR STOP OR CONTINUE SIGNAL
*
*     IF STOP SWITCH ENERGIZED - EXIT DYNAMIC LOOP
      IF(TEST(2).LT.0) GO TO 210
*     IF FLY SWITCH ENERGIZED - CONTINUE DYNAMIC LOOP
  200 IF(TEST(1).LT.0) IFLAG=IFLAG+1 ; GO TO 100
      GO TO 200
  210 CALL STOPCLOCK
      CALL HOLD
      RETURN
      END
```

```
      SUBROUTINE INST
C*************************************************************
C*                                                           *
C*  GENERATES DYNAMIC PORTION OF INSTRUMENT DISPLAY          *
C*                                                           *
C*************************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      COMMON DRVTAB(10,7),CRWDIR(7,17),B
      CCMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90),ASCALE
      COMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      COMMON /GDATA/ YOVSI,DVSI,XPVSI,XODVI,YODVI,DDVI,YOALT,DALT,XPALT,
     1               XOHDG,RP,XCEN,YCEN,YOTURN,XOTURN,XOSLIP,YOSLIP,
     2               YOSPD,DSPD,XPSPD,DTURN,SINPHI,COSPHI,HGT,BASE,
     3               XLEFT,XRIGHT,YTOP,YBOT,DCMPS,YOHDG,RB,EPS,INDYM(83)
      DIMENSION ICOMP(5),IVSIP(5),IDVIL(4),IALTP(5),ICMPS(38),IBANK(5),
     1          IHBAR(2),IPBAR(2),ITURNI(5),IBALL(5),ISPDP(5),IHDG(5)
      EQUIVALENCE (IVSIP(1),INDYM(2)), (IDVIL(1),INDYM(7)), (IALTP(1),
     1            INDYM(11)), (ICMPS(1),INDYM(16)), (IHDG(1),INDYM(54)),
     2            (IBANK(1),INDYM(59)), (IHBAR(1),INDYM(64)), (IPBAR(1),
     3            INDYM(66)), (ITURNI(1),INDYM(68)), (IBALL(1),
     4            INDYM(73)), (ISPDP(1),INDYM(78))
C*
C*  VERTICAL SPEED INDICATOR - POINTER
C*
      Y=YOVSI-DVSI*(AMAX(AMIN(VZ,16.7),-16.7))/4.17
      IVSIP(1)=IPACK(XPVSI,Y,0)
      X=XPVSI+HGT
      Y=Y-BASE/2.
      IVSIP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      IVSIP(3)=IPACK(X,Y,1)
      Y=Y-BASE/2.
      IVSIP(4)=IPACK(XPVSI,Y,1)
      IVSIP(5)=IPACK(X,Y,1)
C*
C*  DIRECTION VELOCITY INDICATOR - SPEED INDICATOR LINES
C*
      Y=YODVI-DDVI*(AMAX(AMIN(VX,23.7),-23.7))/5.93
      IDVIL(1)=IPACK(XLEFT,Y,0)
      IDVIL(2)=IPACK(XRIGHT,Y,1)
      X=XODVI-DDVI*(AMAX(AMIN(VY,23.7),-23.7))/5.93
      IDVIL(3)=IPACK(X,YTOP,0)
      IDVIL(4)=IPACK(X,YBOT,1)
C*
C*  CHECK FOR DESIRED DISPLAY
```

```
*
      IF(TEST(6).GE.0)  NBLK=3 ; GO TO 400
      NBLK=9
*
*   RADAR ALTIMETER - POINTER
*
      IF(ZE.GT.100.) GO TO 200
      Y=YOALT+DALT*(AMAX(ZE,-10.))/5.
      GO TO 210
  200 IF(ZE.GT.200.) GO TO 205
      Y=YOALT+DALT*(20.+(ZE-100.)/25.)
      GO TO 210
  205 Y=YOALT+DALT*(24.+(AMIN(ZE,550.)-200.)/100.)
  210 IALTP(1)=IPACK(XPALT,Y,0)
      X=XPALT+HGT
      Y=Y-BASE/2.
      IALTP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      IALTP(3)=IPACK(X,Y,1)
      Y=Y-BASE/2.
      IALTP(4)=IPACK(XPALT,Y,1)
      IALTP(5)=IPACK(X,Y,1)
*
*   COMPASS HEADING
*
      PSIDEG=PSI*57.3
      IF(PSIDEG.LE.0.)  PSIDEG=PSIDEG+360.
      IPSI=5*(INT(PSIDEG/5.+.5))
      K=(72*IPSI)/360
*     IF HEADING REMAINS CONSTANT DO NOT REGENERATE TEXT
      IF(K.EQ.KSAV) GO TO 250
      KSAV=K
      ENCODE(20,345,ICOMP) (CMPNUM(J),J=K,K+18)
  345 FORMAT(19A1)
      CALL TEXTO(IDEV,ICOMP,5,5,30,2,3,IER)
*
*   CCMPASS - SCALE
*
      L=2*(K/2)
      D1=.10*SCALE
      D2=.25*SCALE
      IF(K.NE.L)  D1=.25*SCALE ; D2=.10*SCALE
      X=-.45
      Y=.882
      ICMPS(1)=IPACK(X,Y,0)
      DO 240  J=2,34,4
        Y=Y+D1
        ICMPS(J)=IPACK(X,Y,1)
```

```
         X=X+DCMPS
         Y=Y-D1
         ICMPS(J+1)=IPACK(X,Y,0)
         Y=Y+D2
         ICMPS(J+2)=IPACK(X,Y,1)
         X=X+DCMPS
         Y=Y-D2
         ICMPS(J+3)=IPACK(X,Y,0)
  240    CONTINUE
       Y=Y+D1
       ICMPS(38)=IPACK(X,Y,1)
*
*   COMPASS - HEADING POINTER
*
  250 XOHDG=(PSIDEG-IPSI)*DCMPS/2.5
      IHDG(1)=IPACK(XOHDG,YOHDG,0)
      X=XOHDG-BASE/2.
      Y=YOHDG-HGT
      IHDG(2)=IPACK(X,Y,1)
      X=X+BASE
      IHDG(3)=IPACK(X,Y,1)
      IHDG(4)=IPACK(XOHDG,YOHDG,1)
      IHDG(5)=IPACK(XOHDG,Y,1)
*
*   VERTICAL GYRO INDICATOR - ANGLE OF BANK POINTER
*
      XP=XCEN-RP*SINPHI
      YP=YCEN+RP*COSPHI
      IBANK(1)=IPACK(XP,YP,0)
      X=XCEN-RB*SIN(PHI+EPS)
      Y=YCEN+RB*COS(PHI+EPS)
      IBANK(2)=IPACK(X,Y,1)
      X=XCEN-RB*SIN(PHI-EPS)
      Y=YCEN+RB*COS(PHI-EPS)
      IBANK(3)=IPACK(X,Y,1)
      IBANK(4)=IPACK(XP,YP,1)
      X=XCEN-RB*SINPHI
      Y=YCEN+RB*COSPHI
      IBANK(5)=IPACK(X,Y,1)
*
*   VERTICAL GYRO INDICATOR - HORIZON BAR
*
      PITCH=-(AMAX(AMIN(THETA,.87),-.87))
      RO=PITCH*ASCALE
      YO=YCEN+RO*COSPHI
      RO=ABS(RO)
      XO=XCEN-RO*SINPHI
      DL=SQRT(RB*RB-RO*RO)
```

```
      X=X0-DL*COSPHI
      Y=Y0-DL*SINPHI
      IHBAR(1)=IPACK(X,Y,0)
      X=X0+DL*COSPHI
      Y=Y0+DL*SINPHI
      IHBAR(2)=IPACK(X,Y,1)
*
*   VERTICAL GYRO INDICATOR - PITCH BAR
*
      R0=(PITCH+.1745)*ASCALE
      Y0=YCEN+R0*COSPHI
      R0=ABS(R0)
      X0=XCEN+R0*SINPHI
      DL=.80*SCALE
      X=X0-DL*COSPHI
      Y=Y0-DL*SINPHI
      IPBAR(1)=IPACK(X,Y,0)
      X=X0+DL*COSPHI
      Y=Y0+DL*SINPHI
      IPBAR(2)=IPACK(X,Y,1)
*
*   RATE OF TURN - INDICATOR
*
      DS=.04
      DL=.08
      Y=Y0TURN-.016
      X=X0TURN-DS/2.+DTURN*(AMAX(AMIN(PSIDOT,.105),-.105))/.0523
      ITURNI(1)=IPACK(X,Y,0)
      X=X+DS
      ITURNI(2)=IPACK(X,Y,1)
      Y=Y-DL
      ITURNI(3)=IPACK(X,Y,1)
      X=X-DS
      ITURNI(4)=IPACK(X,Y,1)
      Y=Y+DL
      ITURNI(5)=IPACK(X,Y,1)
*
*   SLIP INDICATOR - BALL
*
      DS=.18*SCALE
      X=X0SLIP-DS/2.+.862*(PHI-ATAN(U*PSIDOT/32.2))
      Y=Y0SLIP+DS/2.
      IBALL(1)=IPACK(X,Y,0)
      X=X+DS
      IBALL(2)=IPACK(X,Y,1)
      Y=Y-DS
      IBALL(3)=IPACK(X,Y,1)
      X=X-DS
```

```
      IBALL(4)=IPACK(X,Y,1)
      Y=Y+DS
      IBALL(5)=IPACK(X,Y,1)
*
*   AIRSPEED - SCALE POINTER
*
      UK=AMAX(AMIN(UKTS,105.),0.)
      IF(UK.GT.20.) GO TO 300
      Y=YOSPD+DSPD*UK/10.
      GO TO 305
  300 IF(UK.GT.100.) UK=100.
      Y=YOSPD+DSPD*(2.+(UK-20.)/5.)
  305 ISPDP(1)=IPACK(XPSPD,Y,0)
      X=XPSPD-HGT
      Y=Y-BASE/2.
      ISPDP(2)=IPACK(X,Y,1)
      Y=Y+BASE
      ISPDP(3)=IPACK(X,Y,1)
      Y=Y-BASE/2.
      ISPDP(4)=IPACK(XPSPD,Y,1)
      ISPDP(5)=IPACK(X,Y,1)
      GO TO 420
  400 DO 410 I=11,82
        INDYM(I)=0
  410   CONTINUE
  420 CALL GRAPHO(IDEV,INDYM,83,NBLK,IER)
      IF(IER.NE.0) OUTPUT(6) 'ERROR - INDYM',IER
      RETURN
      END
```

```
      SUBROUTINE CREW
*****************************************************
*                                                   *
*  GENERATE APPROPIRATE DIRECTIONS FROM CREW        *
*                                                   *
*****************************************************
      INTEGER CRWDIR,DRVNAM,ALTNUM,AIRNUM,CMPNUM
      COMMON DRVTAB(10,7),CRWDIR(7,17),B
      COMMON /GRAPH/ IDEV,SCALE,ITD(25),IGD(25),NBLK,NULL,IBLANK(2),
     1               ALTNUM(18),AIRNUM(20),CMPNUM(90)
      COMMON /STATE/ U,V,W,UKTS,P,Q,R,VX,VY,VZ,THETA,PHI,PSI,THEDOT,
     1               PHIDOT,PSIDOT,XE,YE,ZE,DT,FLTIM,VXRMS,VYRMS,VZRMS,
     2               XERMS,YERMS,ZERMS,JFLAG,KFLAG,RMSTIM,KSAV
      AXE=ABS(XE)
      IF(XE.GE.-1100.) GO TO 705
      I=17
      J=1
      IX=NULL
      IY=NULL
      HOVTIM=0.
      GO TO 765
  705 IF(XE.GE.-500.) GO TO 710
      I=1
      J=16
      AXE=200.*INT(AXE/200.+.5)
      ENCODE(4,730,IX) AXE
      HOVTIM=0.
      GO TO 765
  710 IF(J.EQ.13) GO TO 755
      IF(J.EQ.14) GO TO 765
      IF(AXE.LE.3.)  I=4 ; IX=NULL ; GO TO 735
      IF(XE.GE.3.)   GO TO 715
      IF(XE.LT.-30.) I=2 ; GO TO 720
      IF(VX.LT.B*AXE) I=2 ; GO TO 720
      I=3
      IX=NULL
      GO TO 735
  715 IF(XE.GT.60.)  I=7 ; J=8 ; IX=NULL ; IY=NULL ; GO TO 765
      IF(XE.GT.20.)  I=6 ; GO TO 720
      IF(VX.GT.-B*AXE) I=6 ; GO TO 720
      I=5
      IX=NULL
      GO TO 735
  720 IF(AXE.GE.100.) AXE=100.*INT(AXE/100.+.5) ; GO TO 725
      IF(AXE.GT.10.)  AXE=10.*INT(AXE/10.+.5)   ; GO TO 725
      AXE=INT(AXE+.5)
  725 ENCODE(4,730,IX) AXE
  730 FORMAT(I4)
```

```
*
*   CHECK LATERAL POSITION
*
  735 AYE=ABS(YE)
      IF(AYE.LE.3.)   J=16 ; IY=NULL ; GO TO 752
      IF(I.EQ.4)      I=16
      IF(YE.LT.-3.)   GO TO 740
      IF(YE.GT.25.)   J=9 ; GO TO 745
      IF(VY.GT.-B*AYE) J=9 ; GO TO 745
      J=10
      IY=NULL
      GO TO 752
  740 IF(YE.LT.-25.) J=11 ; GO TO 745
      IF(VY.LT.B*AYE) J=11 ; GO TO 745
      J=12
      IY=NULL
  745 IF(AYE.GT.10.) AYE=10.*INT(AYE/10.+.5) ; GO TO 747
      AYE=INT(AYE+.5)
  747 ENCODE(4,730,IY) AYE
*
*   COMPUTE HOVER TIME
*
  752 IF(JFLAG.EQ.1) GO TO 754
  753 IF(I.EQ.4.AND.J.EQ.16) JFLAG=1 ; KFLAG=1 ; GO TO 755
      HOVTIM=0.
      GO TO 765
  754 IF(AXE.GT.10..OR.AYE.GT.10.) JFLAG=0 ; HOVTIM=0. ; GO TO 765
  755 HOVTIM=HOVTIM+DT
      IF(HOVTIM.GT.150.) I=4 ; J=14 ; KFLAG=0 ; GO TO 765
      IF(HOVTIM.GT.120.) I=4 ; J=13 ; GO TO 765
*
*   CHECK ALTITUDE
*
  765 IF(ZE.LE.15.) I=15 ; J=16 ; GO TO 775
*
*   CUTPUT CREW DIRECTIONS
*
  770 CRWDIR(5,I)=IX
      CRWDIR(5,J)=IY
  775 CALL TEXTO(IDEV,CRWDIR(1,I),6,35,29,2,3,IER)
      CALL TEXTO(IDEV,CRWDIR(1,J),6,37,29,2,3,IER)
      RETURN
      END
```

APPENDIX C

DIGITAL COMPUTER PROGRAM FORTRAN VARIABLES

ABS	Absolute value -- intrinsic subprogram.
ADK	External subprogram used to perform analog to digital conversion.
AIRNUM	Airspeed numbers -- array containing the numbers for the airspeed scale.
ALTNUM	Altimeter numbers -- array containing the numbers for the radar altimeter scale.
AMAX	Maximum value of two arguments -- intrinsic subprogram.
AMIN	Minimum value of two arguments -- intrinsic subprogram.
AROLL	Absolute value of ROLL.
ASCALE	Angle scale -- scale factor for converting an angle to a linear displacement.
ASI	Airspeed indicator -- scaled value of airspeed sent to cockpit indicator.
ATAN	Arctangent -- intrinsic subprogram.
AXE	Absolute value of XI.
AYE	Absolute value of YE.
BASE	Base -- length of the base of the triangular pointers used in the integrated display.
BETA	Sideslip angle.
CMPNUM	Compass numbers -- array containing the numbers for the compass scale.
COMPUTE	External subprogram used to place the analog computer in the "compute" mode.

COS	Cosine -- intrinsic subprogram.
COSPHI	Cosine of PHI.
COSPSI	Cosine of PSI.
COSTHE	Cosine of THETA.
CREW	Crew -- subprogram which generates directions from a simulated rescue aircrewman.
CRWDIR	Crew directions -- array containing crew directions.
D1	Length of a scale mark on the compass heading scale.
D2	Length of a scale mark on the compass heading scale.
DA1C	A_{1c} -- change in lateral cyclic, radians.
DAC	External subprogram used to perform digital to analog conversion.
DALT	Altimeter division -- distance between divisions of the radar altimeter scale.
DCMPS	Compass division -- distance between divisions of the compass heading scale.
DDVI	Direction velocity indicator division -- distance between divisions of the direction velocity indicator scale.
DGINIT	Graphics initialization subroutine.
DL	Long displacement -- length of a long scale mark.
DRVNAM	Derivative name -- array containing the names of the stability derivatives.
DRV	Derivative -- array containing the airspeed dependent stability derivaties for a specified airspeed.
DRVTAB	Derivative table -- array containing the airspeed dependent stability derivaties for several airspeeds.
DS	Short displacement -- length of a short scale mark.

DSPD	Speed division -- distance between divisions of the airspeed scale.
DSPLY	Display -- subprogram which generates the static portions of the integrated instrument display.
DT	Time interval.
DTHR	θ_R -- change in tail rotor pitch, radians.
DTHRC	Value of θ_R required to maintain zero side-slip flight.
DTINIT	Text initialization rubroutine.
DTURN	Turn division -- distance between marks of the turn indicator.
DVSI	Vertical speed indicator division -- distance between divisions of the vertical speed indicator.
EPS	Small angle.
FLTIM	Flight time.
FLY	Fly -- subprogram which generates information for and controls the solution of the helicopter dynamics.
GCOSPH	Factor in Z-Force equation.
GRAPHO	Graphics output -- external subprogram, used to output a graphics array to the graphics processor.
GSINPH	Factor in Y-Force equation.
GSINTH	Factor in X-Force equation.
HGT	Height -- height of the triangular pointers used in the integrated display.

HOLD	External subprogram used to place the analog computer in the "hold" mode.
HOVTIM	Hover time -- elapsed time within a specified distance from the target.
I	Integer counter.
IALT	Altimeter -- graphics data array for the radar altimeter scale.
IALTP	Altimeter pointer -- graphics data array for the radar altimeter pointer.
IBALL	Ball -- graphics data array for the slip indicator ball.
IBANK	Bank angle -- graphics data array for the attitude gyro angle of bank pointer.
IBLANK	Blank -- graphics data array used to blank out another graphics data array.
ICMPS	Compass -- graphics data array for the compass heading scale.
ICOMP	Compass -- text array for the compass heading numbers.
ICOPS	Initial character position -- fixes the lateral position on the graphics display of the first character in a text array.
IDEV	Device number -- the number 1 or 2 which specifies the graphics processor to be used.
IDVI	Direction velocity indicator -- graphics data array for the direction velocity indicator scales.
IDVIL	Direction velocity indicator lines -- graphics data array for the direction velocity indicator speed lines.
IER	Error parameter returned after calls to DGINIT, DTINIT, GRAPHO or TEXTO.
IFLAG	Integer counter -- counts number of times through dynamic loop in FLY.
IGD	Graphics directory -- argument of DGINIT.

IHBAR	Horizon bar -- graphics data array for the attitude gyro artificial horizon line.
IHEAD	External subprogram used to generate the first word of a graphics array.
IHDG	Heading -- graphics data array for the compass heading pointer.
IMAP	Miniature airplane -- graphics data array for the attitude gyro miniature airplane reference.
INDYM	Instrument dynamics -- graphics data array for the moving (dynamic) portions of the instrument display.
INT	Converts a number to an integer -- intrinsic subprogram.
INST	Instrument -- subprogram which generates the dynamic portions of the integrated display.
IPACK	External subprogram used to generate words of a graphics array.
IPSI	PSI converted to integer value.
IPBAR	Pitch bar -- graphics data array for the attitude gyro pitch line.
ISLIP	Slip -- graphics data array for the slip indicator center marks.
ISPD	Speed -- graphics data array for the airspeed scale.
ISPDP	Speed pointer -- graphics data array for the airspeed pointer.
ITD	Text directory -- argument of DTINIT.
ITURN	Turn -- graphics data array for the turn indicator scale.
ITURNI	Turn indicator -- graphics data array for the turn needle.
IVGI	Vertical gyro indicator -- graphics data array for the attitude gyro angle of bank scale.
IVSI	Vertical speed indicator -- graphics data array for the vertical speed indicator scale.

IVSIP — Vertical speed indicator pointer -- graphics data array for the vertical speed indicator pointer.

IX — Integer X -- integer value of XE.

IY — Integer Y -- integer value of YE.

J — Integer counter.

JFLAG — Integer flag used to control accumulation of hover time.

K — Integer counter

K2 — Integer value retained for later comparison.

KFLAG — Integer flag used to control accumulation of RMS performance parameters.

KSAV — K save -- saves value of K for later comparison.

L — Integer counter.

LN — Line number -- specifies line position of a text block.

LNO — Same as LN except refers to initial line.

LVU — $L_V(u)$ -- partial derivative of rolling moment.

MAU — $M_A(u)$ -- partial derivative of pitching moment.

MB1CU — $M_{B_{1c}}(u)$ -- partial derivative of pitching moment.

NBLK — Block number -- refers to graphics data blocks.

NGD — Number of words in the graphics directory -- argument of DGINIT.

NRU — $N_R(u)$ -- partial derivative of yawing moment.

NTD	Number of words in the text directory -- argument of DTINIT.
NULL	Null -- text array of blank spaces.
NVU	$N_v(u)$ -- partial derivative of yawing moment.
P	P -- pitch rate.
PHI	$\emptyset$-- roll angle.
PHIDOT	$\dot{\emptyset}$
PHIDTS	$\dot{\emptyset}$ scaled for the analog computer.
PITCH	θ limited to $\pm$ 50°.
POTSET	Subprogram which places the analog computer in the POTSET mode.
PSI	ψ -- yaw angle
PSIDEG	ψ scaled to degrees.
PSIDOT	$\dot{\psi}$
PSIDTS	$\dot{\psi}$ scaled for the analog computer.
Q	q -- roll rate
R	r -- yaw rate.
RO	Initial radius -- radial distance from the center of the attitude gyro to the angle of bank scale marks.
RADALT	Radar altimeter -- scaled value of altitude sent to cockpit radar altimeter.
RB	Radius to base -- radial distance from the center of the attitude gyro to the base of the triangular angle of bank pointer.
READCLOCK	External subprogram used to read the present value of the analog computer clock.

RESET — Reset -- subprogram which places analog computer in Reset mode.

RL — Radial line -- length of the radial line segment used for the attitude gyro angle of bank scale marks.

RMSTIM — Root mean square time -- time interval used to compute performance parameters.

ROLL — Roll -- angular position of the attitude gyro angle of bank scale marks.

RP — Radius to point -- radial distance from the center of the attitude gyro to the point of the angle bank pointer.

SETLINES — External subprogram used to set analog computer logic.

SETPOT — External subprogram used to set the analog computer potentiometers.

SCALE — Scale -- multiplying factor to convert $\pm$ 5 inches to $\pm$ 1 units for graphics processor.

SIN — Sine -- intrinsic subprogram.

SINPHI — Sine of PHI.

SINPSI — Sine of PSI.

SINTHE — Sine of THETA.

SQRT — Squre root -- intrinsic subprogram.

STARTCLOCK — External subprogram used to start the analog computer clock.

STOPCLOCK — External subprogram used to stop the analog computer clock.

TEST — External subprogram used to test the logic of specified analog trunk lines.

TEXTO — External subprogram used to output a text array to the graphics processor.

THEDOT — $\dot{\theta}$

THEDTS	$\dot{\theta}$ scaled for the analog computer.
THETA	θ -- pitch angle.
THETIC	Initial condition on θ.
TNEW	New time.
TOLD	Old Time.
U	u -- forward velocity.
U1	Value of airspeed used in the linear interpolation subroutine.
U2	Value of airspeed used in the linear interpolation subroutine.
UK	u scaled to knots.
UKTS	u scaled to knots.
V	v -- lateral velocity.
V1	Intermediate calculation for VX and VY.
V2	Intermediate calculation for VX and VY.
VX	Inertial velocity along the x-axis.
VXRMS	Root mean square value of VX. Used as a performance parameter.
VY	Inertial velocity along the y-axis.
VYRMS	Root mean square value of VY. Used as a performance parameter.
VZ	Inertial velocity along hte z-axis.
VZRMS	Root mean square value of VZ. Used as a performance parameter.
VZS	VZ scaled for the analog computer.
W	w -- vertical velocity.
WRITECLOCK	External subprogram used to set the analog computer clock to a specified value.

X	x coordinate position used for graphics construction.
XO	X initial -- initial x coordinate position used for graphics construction.
XOALT	XO for the radar altimeter scale.
XODVI	XO for the radar altimeter scale.
XODVI	XO for the direction velocity indicator scale.
XOHDG	XO for the compass heading pointer.
XOSLIP	XO for the slip indicator scale.
XOSPD	XO for the airspeed scale.
XOTURN	XO for the turn rate scale.
XOVSI	XO for the vertical speed scale.
XAU	$X_A(u)$ -- aerodynamic force in the X direction.
SCEN	x center -- x coordinate position for the center of the attitude gyro.
XE	x earth -- x coordinate position of the helicopter in the inertial reference axes.
XERMS	XE root mean square -- used as a performance parameter.
XLEFT	x left -- x coordinate position of the left end of the direction velocity indicator speed line.
XP	x pointer -- x coordinate position for the point of a scale pointer.
XPALT	XP for the radar altimeter scale pointer.
XPSPD	XP for the airspeed scale pointer.
XPVSI	XP for the vertical speed scale pointer.
XRIGHT	x right -- x coordinate position for the right end of the direction velocity indicator speed line.
XTHCU	$X_{\theta_c}(u)$ -- partial derivative of X force.

Y	y coordinate position used for graphics construction.
YO	y initial -- initial x coordinate position used for graphics construction.
YOALT	YO for the radar altimeter scale.
YODVI	YO for the direction velocity indicator scale.
YOHDG	YO for the compass heading pointer.
YOSLIP	YO for the slip indicator scale.
YOSPD	YO for the airspeed scale.
YOTURN	YO for the turn rate scale.
YOVSI	YO for the vertical speed scale.
YBOT	y bottom -- y coordinate position for the bottom end of the direction velocity indicator speed line.
YCEN	y center -- y coordinate position for the center of the attitude gyro.
YE	y earth -- y coordinate position of the helicopter in the inertial reference axes.
YERMS	YE root mean square -- used as a performance parameter.
YP	y pointer -- y coordinate position for the point of a scale pointer.
YTOP	y top -- y coordinate position for the top end of the direction velocity indicator speed line.
ZAU	$Z_A(u)$ -- aerodynamic force along z axis.
ZB1CU	$Z_{B_{1c}}(u)$ -- partial derivative of z force.
ZE	z earth - z coordinate position for the helicopter in the inertial reference axes (altitude).

ZEIC	ZE initial condition -- starting value of ZE.
ZERMS	ZE root mean square -- used as a performance parameter.
ZES	ZE scaled for the analog computer.
ZWU	$Z_W(u)$ -- partial derivative of z force.

APPENDIX D

HELICOPTER SIMULATION SYSTEM CHECKLIST

ANALOG SET-UP

1. Install Patch Boards (#8) on CI-5000 (HANDLES UP)
2. Turn CI-5000 on
3. Set POT 400 to +20.00 Volts
4. Set POT 401 to +20.00 Volts
5. Set POT 437 to +30.00 Volts
6. Set Limiters L00 and L07 to $\pm$ 1.0 Volts
7. Center all Digital Function Switches
8. Press "DIGITAL CMPTR" Mode Switch

AGT-10 SET-UP

1. Turn on XDS-9300
2. Place "OLD AMOS" discs on appropriate AGT-10 disc drive
3. Turn Disc Drive on - "READY" light should come on
4. Turn on "THIS IS IT" switch at back of AGT-10 Mainframe
5. Press HALT, RESET, RUN, PULSE 1 on OCP
6. "MO/DA/YR" should appear at teletype - if not, follow bootstrap loading instructions attached to AGT-10 OCP
7. Type 7/7/77, press "return"
8. Type RESET ("GATED", 101)!
9. After TTY returns, type GATED!

10. To check that GATED is properly loaded, press upper left function switch on the function switch console. (The message "TEXT BLOCK SELECT MODE BLOCK 1" should appear on the lower edge of the screen.)

XDS-9300 SET-UP

1. Load "HELO SIMULATION" tape on either of the two tape drives
2. Run tape forward to "LOAD POINT"
3. Set Mode Selector switch to "AUTOMATIC"
4. Set Tape Unit Selector switch to "1"
5. Place tape rerun and data cards in card reader
6. Press "POWER ON" and "START" on card reader ("NOT READY" light should go out)
7. Press "READY" button on line printer if ready light not on
8. Select "EXT" on XDS-9300 clock switch
9. Press "RESET", "RUN", "CARDS", on XDS-9300 control console
10. When teletypewriter message is received, select appropriate AGT-10 by typing "IDEV = 1 or 2*" and press "RETURN" key
11. Set up TV camera

COCKPIT SET-UP

1. Set "FLY" switch in down position
2. Set instrument display switch to "INTEGRATED"
3. Turn on "MASTER POWER", "FLIGHT SYSTEM", and "DC POWER SUPPLY" switches at rear of cockpit

4. Close latching mechanism on cockpit terminal patch board
5. Turn on TV repeater

LIST OF REFERENCES

1. Stein, K. J., "Helicopter Control Concepts Emerging," Aviation Week and Space Technology, v. 103, no. 15, p. 44-51, 13 October 1975.
2. Sweeney, C. J., Conversion of AC-11B Instrument Flight Trainer into a Variable Stability Flight Simulator, Masters Thesis, Naval Postgraduate School, Monterey, California, September 1968.
3. Huckemeyer, M. R., Incorporation of the Six Dimensional Linearized Equations of Motion with a Variable Stability Flight Simulator, Masters Thesis, Naval Postgraudate School, Monterey, California, September 1971.
4. Hoxie, S. S., The Implementation of a Fixed Base Helicopter Simulation in the Investigation of an Automatic Scan System, Engineers Thesis, Naval Postgraduate School, Monterey, California, March 1974.
5. Ammerman, L. R., Evaluation of an Integrated Electronic Instrument Display for Helicopter Hover Operations Using a Six-Degree-of-Freedom Fixed-Base Simulation, Masters Thesis, Naval Postgraduate School, Monterey, California, March 1975.
6. Naval Postgraduate School Computer Laboratory, Manual for the Fortran User, Naval Postgraduate School, Monterey, California, 1975.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 67 Department of Aeronautics Naval Postgraduate School Monterey, California 93940	1
4.	Associate Professor D. M. Layton, Code 67Ln Department of Aeronautics Naval Postgraduate School Monterey, California 93940	3
5.	LT William Woodrow Fetzer, Jr., USN 305 Center Street Millville, Pennsylvania 17846	1

Thesis
F2695
c.1

171784

Fetzer

Evaluation of and operational procedures for a helicopter simulation system utilizing an integrated electronic instrument display.